REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-00-

0486

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for r~~~~~ ~aintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estima~~~~~ ~tions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jeffer~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE 01 Sept 2000	3. REPORT TYPE AND DATES COVERED Final 15 Jan 1998--30 Nov.2000	
4. TITLE AND SUBTITLE Nonlinear Modeling: Estimation and Prediction with Applications to GPS		**5. FUNDING NUMBERS** G AFOSR-F49620-98-1-0190	
6. AUTHOR(S) Hamid Krim			

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Department of Electrical and Computer Engineering NC State University Campus Box 7911 Raleigh, NC 27695	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Dr. Jon Sjogren, c/o Dr. Mark Gruneisen 801 Randolph Street, Room 732 Arlington, VA 22203-1977	10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

Nonlinear Scale-based signal and analysis techniques are widely viewed as a viable alternative eo

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release, distribution unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

Nonlinear Scale-based signal and analysis techniques are widely viewed as a viable alternative to meet the challenges of increasingly complex engineering problems and are explored in this research. The research effort described in this report addresses two problems which require such techniques:

1. To make Global Positioning System more robust to outside interference and perturbations.

2. To perform filtering on images for Automatic Target Recognition driven by features which may be potential signatures.

20001002 010

14. SUBJECT TERMS			15. NUMBER OF PAGES 63
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

Final Report on the Grant
AFOSR-F49620-98-1-0190
Nonlinear Modeling: Estimation and Prediction with Applications to GPS
01/15/98 to 11/30/00

Hamid Krim
Program Director: Dr. Jon Sjogren

August 29, 2000

Contents

List of Figures

4

Abstract

Nonlinear Scale-based signal and analysis techniques are widely viewed as a viable alternative to meet the challenges of increasingly complex engineering problems. The research effort described in this report addresses two problems which require such techniques:

1. To make Global Positioning System more robust to outside interference and perturbations

2. To perform filtering on images for Automatic Target Recognition driven by features which may be potential signatures.

Chapter 1

Summary of contributions

The funding of AFOSR-F49620-98-1-0190 grant has helped the PI establish a solid footing for a new research program, and has resulted in substantial contributions in Nonlinear Multi-scale Estimation and Filtering. A Master's thesis funded at 100% authored by B. Karacali was completed [1] and two partially (50% funding each) supported Ph.D. theses by Y. F. Bao (May'01) and O.V. Poliannikov (May 02) are near completion . Partial support of this grant has also led to the publication of four journal papers [2, 3, 4, 5] and three journal papers under review [6, 7, 8], as well as two invited book chapters[9, 10] and several conference papers at ICASSP and ICIP [11, 12, 13, 14] and invited national and international seminars, including an invitation for a one month visit in the department of information sciences at University of Tokyo (Japan) (Nov. 1999), courtesy of the Japanese Society for Advancement of Science, and an invitation as principal guest speaker at a workshop on Mathematics in Imaging at University of Hong Kong (China) (Dec. 2000).

In the course of this project, working contacts have also been initiated with key technical members of the GPS group of the Air Force Laboratories. Our plan in the very near future is to contact the ATR groups around the Air Force Laboratories and to better communicate some of our results and help fine tune them to specific problems of direct interest to the Air Force.

The increasing complexity of engineering problems in general, together with the demand for additional performance in signal and image processing problems in particular, have led to an intensification of search for nonlinear alternative approaches. This is widely viewed as a promising framework which would provide reliable as well as robust techniques to meet new challenges.

Towards that goal, the work performed over the duration of the current grant has focused on multi-scale nonlinear techniques with two main application thrusts:

1. Robust methods for Global Positioning System applications

2. Automatic Target Recognition in Synthetic Aperture Radar, Infra-red and Inverse Synthetic Aperture Radar imaging.

In GPS, a frequently encountered important problem in particularly adverse conditions (e.g. intentional jamming) is a loss of lock which in turn, may lead to unexpected delays and hence to serious consequences in critical GPS applications.

The first part of this report addresses such a problem, using nonlinear wavelet techniques which are well known to show particular resilience to noisy environments and and to exhibit robustness to perturbations. Specifically, we develop a new code lock detection technique which among others, facilitates the detection of the location of the data bit transition. In the second part of this report, we explore the potential of nonlinear scale-based techniques in ATR. Specifically, we develop a probabilistic framework for well known nonlinear filtering/diffusion techniques which give an intuitively appealing interpretation which we in turn use to develop more performing techniques in blind image enhancement [6] and techniques which use features inherent to the image to drive the filtering.

Part I

Robustness of GPS

Chapter 2

Nonlinear Wavelet Techniques in GPS

2.1   INTRODUCTION

Global Positioning System (GPS) is today's most advanced navigation system, which incorporates a position determination tool through a web of satellites that broadcast their exact position in orbit repetitively at particular time instants. By decoding each satellite's message to get their exact position, and by computing their individual distances to the receiver, a user can determine their position on the earth's surface with very good accuracy.

A successful operation of the receiver consists of its simultaneous decoding of at least four different satellites which in turn requires tracking of the signal carrier frequency and code delay. This requirement, as illustrated in Figure 2.1, is reflected by the demodulation step as well as the receiver's alignment of its Pseudo Random Noise (PRN) sequence to that of the received signal. On account of currently used discriminators' vulnerability to bit transitions, the receiver has to acquire lock in both domains within a 20 m-sec time interval.

The present work addresses this problem and provides additional robustness to carrier frequency and code delay lock discriminators which are subject to such navigation message bit transitions. In what follows, we first summarize the GPS receiver operation, and subsequently describe the robust discriminators together with performance comparisons with the current discriminators using real GPS data.

2.2   GPS OVERVIEW

Figure 2.1 shows the block diagram of a generic GPS receiver. Upon its preconditioning and discretization at a lower intermediate frequency (IF), the radio frequency (RF) signal is fed to receiver channels which track signals from different satellites.
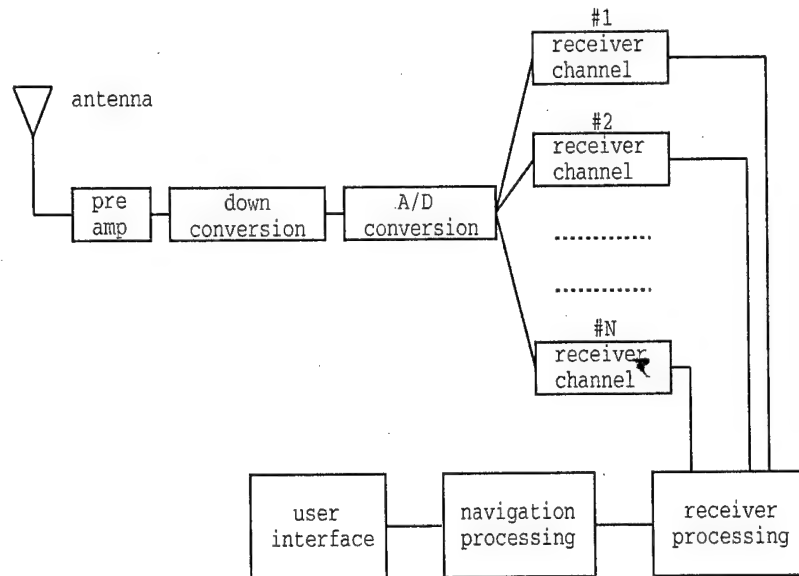
Figure 2.1: The block diagram of a generic GPS receiver

The receiver channels provide the demodulated navigation messages of all observable satellites to the receiver processing, which then feeds the readings to navigation processing that computes the position for the user.

The accurate computation of the position is therefore crucially dependent on accurate demodulation of several satellites' signals, which is intrinsic to the receiver channels.

2.2.1   Generic Receiver Channel

A simplified block diagram of a generic receiver channel is shown in Figure 2.2. The incoming digital IF signal is first stripped off from its carrier through the sine and cosine maps, and then collapsed to baseband once correlated with the receiver generated PRN code.

The receiver processor acquires frequency, phase and code delay errors through I_e, I_p, I_l, Q_e, Q_p and Q_l readings, and tries to preserve the locks by adjusting the frequency and phase of the sine and cosine maps, and the clock rate of the PRN code generator.
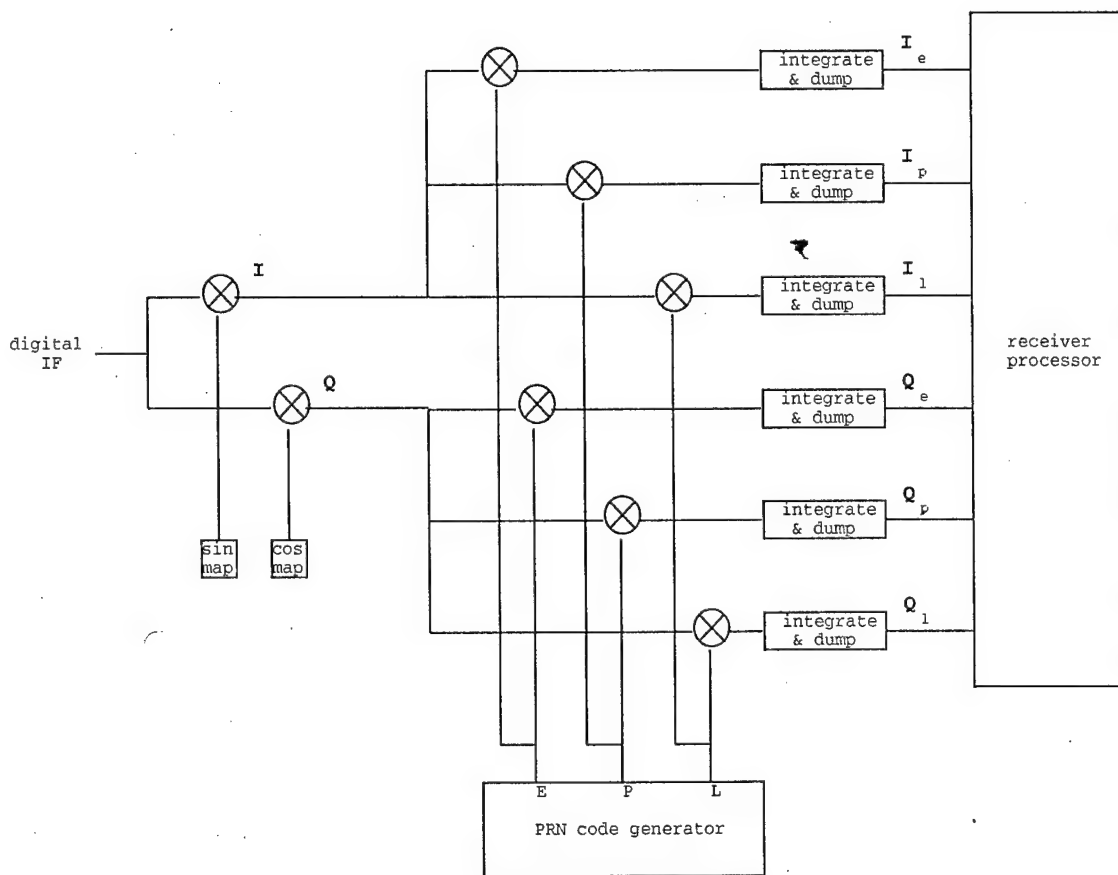
Figure 2.2: The block diagram of a generic GPS receiver channel

2.3 CURRENT LOCK DISCRIMINATORS

The lock discriminators used in this work are summarized below. A more through presentation is available in [15] §5.1.

2.3.1 Frequency and Phase Lock Discriminators

In order to be able to strip the carrier from the spread spectrum signal, the receiver needs to know the precise carrier frequency of the incoming digital IF signal. One of the most commonly used frequency lock discriminators is given by:

$$\delta_f = \frac{\phi_1 - \phi_2}{(t_2 - t_1)2\Pi}$$

where $\phi_i = \text{atan2}(\mathbf{I}_p^i, \mathbf{Q}_p^i)$, and $\mathbf{I}_p^i$ is the $\mathbf{I}_p$ sample observed at time t_i, for $i = 1, 2$. This discriminator gives the true frequency error in a neighborhood inversely proportional to the integration time. The plot of the discriminator output versus true error is shown in Figure 2.3. The dashed line in Figure 2.3 indicates the ideal discriminator output.
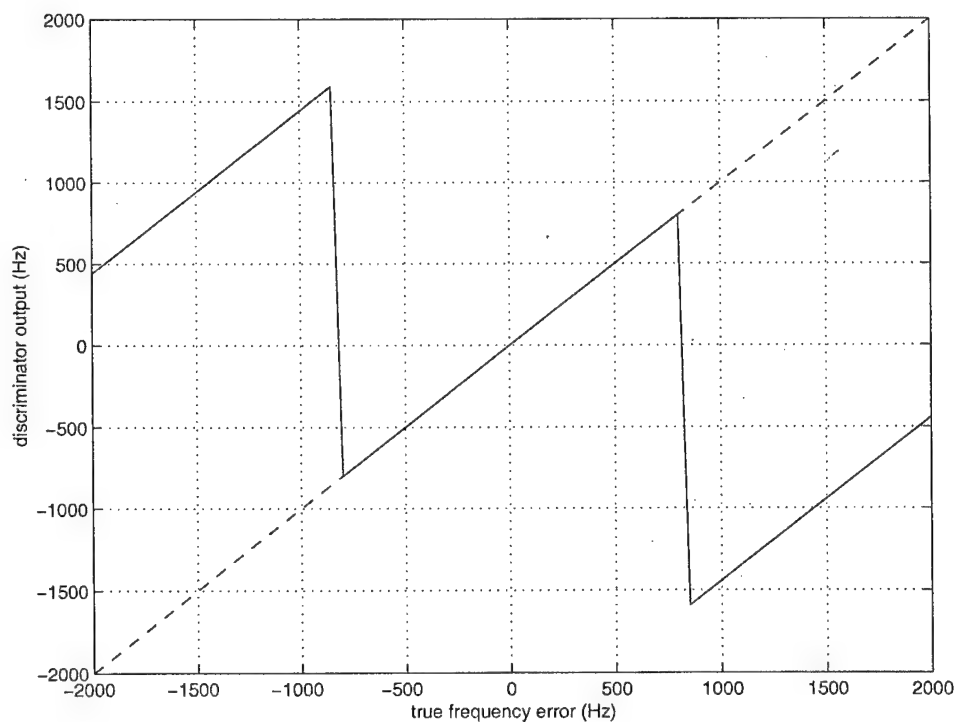
Figure 2.3: The error output of the regular FLL discriminator

The phase lock is generally obtained by a Costas loop. The optimal phase lock discriminator [15] is given by

$$\delta_\phi = \text{atan}(\mathbf{I}_p/\mathbf{Q}_p).$$

7

This phase discriminator gives the true phase error in a neighborhood of $\pm\Pi/2$.

2.3.2   Code Delay Lock Discriminators

Among a number of common delay lock discriminators, the normalized early minus late envelope discriminator given by

$$err = \frac{\sum \sqrt{I_e^2 + Q_e^2} - \sum \sqrt{I_l^2 + Q_l^2}}{\sum \sqrt{I_e^2 + Q_e^2} + \sum \sqrt{I_l^2 + Q_l^2}}$$

provides an amplitude insensitive delay error within $\pm$ 0.5 bit delay. In spite of its high computational load, it will be used in this text as a basis for comparison for the proposed code delay lock discriminator. The theoretical error output of this common discriminator is shown in Figure 2.4.
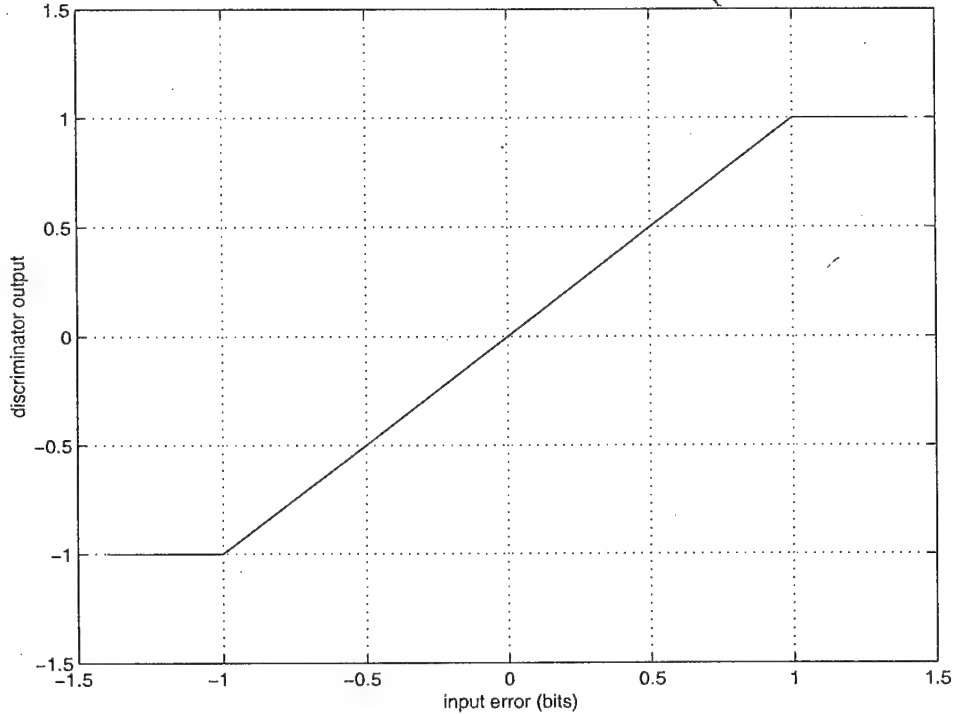
Figure 2.4: The theoretical error output of the normalized power DLL discriminator

2.4   PROPOSED LOCK DISCRIMINATORS

The lock discriminators described here are based on research work presented in [16]. In [16] §5.2.2, an operator $\mathcal{O}_{j_0}$ acting on a signal s that returns high values for signal regions

8

with embedded singularities is defined. An operator d defined through $\mathcal{O}_{j_0}$ with $j_0 = 1$ as

$$d(s) = \max\left\{\max_k\{\mathcal{O}_1(s)\}, \max_k\{\mathcal{O}_1(\tilde{s})\}\right\}$$

where k is the time index of $\mathcal{O}_{j_0}(\cdot)$[1], $s : [0, t_0] \to \mathbb{R}$, and $\tilde{s}(t)$ is given by

$$\tilde{s}(t) = \begin{cases} s(t) & if\, t \in [0, \frac{t_0}{2}] \\ -s(t) & otherwise \end{cases} \quad,$$

is shown to be robust to sign transitions in $s(t)$, which creates potential for use in lock discriminators with robustness to bit transitions.

2.4.1 Robust Frequency Lock Discriminator

Let $\tilde{\mathbf{I}}_p$ denote the input of the integrate and dump operator that produces $\mathbf{I}_p$ samples. Then, the robust frequency loop lock (FLL) discriminator is defined as

$$\delta f_r = \frac{\varphi_1 - \varphi_2}{(t_2 - t_1)2\Pi}$$

where $\varphi_1 = \mathrm{atan2}(I_p^i, Q_p^i)$ with $I_p^i = d(\tilde{\mathbf{I}}_p) \cdot \mathrm{sign}(\sum \tilde{\mathbf{I}}_p)$ observed for a certain period of time until t_i, for $i = 1, 2$. Figure 2.5 shows the discriminator output versus true frequency error to be almost linear with slope 1 around the true frequency, indicated by the dashed line, and therefore, the discriminator indeed can be used to determine the frequency error between the incoming and the receiver generated carriers.

2.4.2 Robust Code Delay Lock Discriminator

The robust code delay lock (CDL) discriminator is defined by the following equation:

$$err_r = \frac{\sqrt{d(s_e)} - \sqrt{d(s_l)}}{\sqrt{d(s_e)} + \sqrt{d(s_l)}}$$

where

$$s_e = \begin{cases} \sqrt{\mathbf{I}_e^2 + \mathbf{Q}_e^2} \cdot \mathrm{sign}(\mathbf{I}_e) & if\, E_I > E_Q \\ \sqrt{\mathbf{I}_e^2 + \mathbf{Q}_e^2} \cdot \mathrm{sign}(\mathbf{Q}_e) & otherwise \end{cases} \quad,$$

$$E_I = \sum \mathbf{I}_e^2 \,,$$

$$E_Q = \sum \mathbf{Q}_e^2 \,.$$

The empirical plot of the error output versus input delay of the robust discriminator is shown in Figure 2.6. Note that the plot is very similar to the theoretical input output response of the regular code delay lock detector, showing linear error within $\pm\, 0.5$ bit of delay error.

---

[1]Note that $k = 1, \dots, 2^{j_0}$.
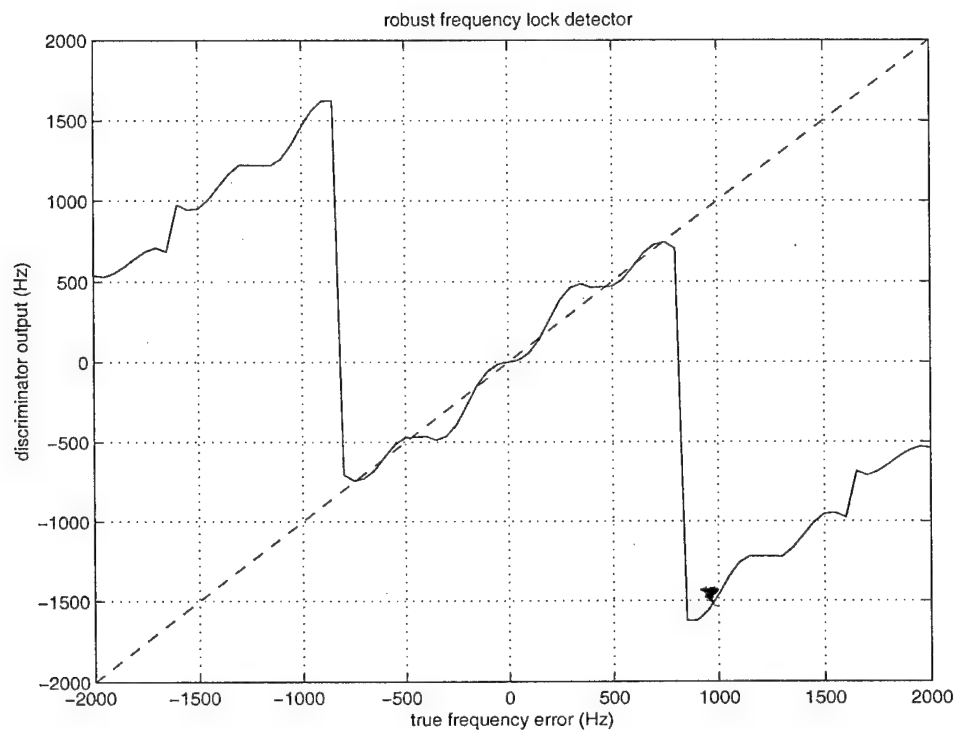
Figure 2.5: The error output of the robust FLL discriminator
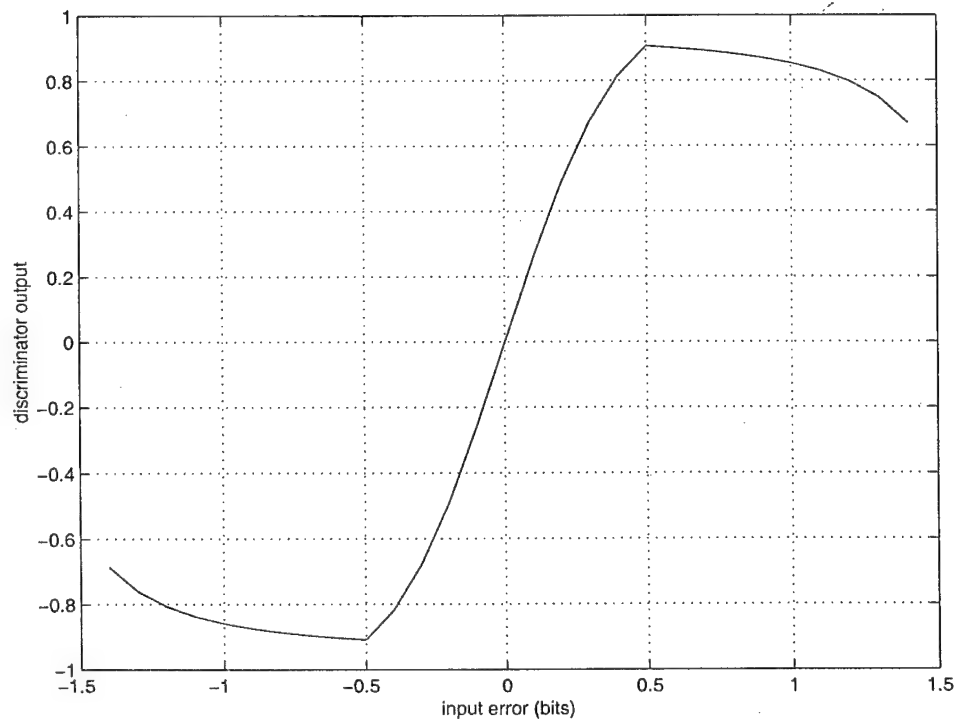
Figure 2.6: The error output of the robust DLL discriminator

2.5 PERFORMANCE ANALYSIS

The performance analysis of the regular lock discriminators and the proposed robust discriminators is conducted using real GPS data[2]. The data consists of the output of the analog-to-digital converter in a GPS receiver, which is the input fed to individual receiver channels for carrier and code stripping processes.

2.5.1 Frequency Tracking Performance

The vulnerability of the regular frequency lock loop discriminators to bit transitions in the navigation signal is well known. This is also demonstrated in Figure 2.7 which shows frequencies at which the regular discriminator indicates the frequency lock. For some time where no bit transitions occur in the data stream, the lock appears stable. Any inadvertent bit transition, however, causes the discriminator to either lock at another frequency, or loose the frequency lock altogether. Note that the spikes which depict the loss of the frequency lock are exactly 20 msec apart, indicating that they indeed correspond to the bit transitions of the 50 Hz navigation signal.

The proposed robust FLL discriminator however, does not show erratic behaviors around the bit transition regions that fool the regular discriminator, maintaining the frequency lock regardless of the navigation message bit transitions, as shown with solid lines in Figure 2.8. The dashed lines in Figure 2.8 represent the regular frequency lock discriminator track.

2.5.2 Code Delay Tracking Performance

In [16], it was empirically shown that the robustness to bit transitions in the conventional code delay lock detectors based on envelope detection was at a cost of a higher sensitivity to noise. The performance comparison between the regular code delay lock discriminators and the proposed discriminators should hence be carried out as a function of prevailing noise.

Towards that end, the real GPS data is demodulated separately using regular and proposed delay lock discriminators, both with the regular frequency lock discriminator, on a signal region with no navigation message bit transitions. This procedure is fifty hundred times, and with various levels of white noise added artificially to the initial real GPS signal. The results are shown in Table 2.1.

The simulation results indicate that in similar conditions, the proposed delay lock discriminator offers better noise performance than the regular discriminator, under additive white Gaussian noise. The increasing noise levels impair both discriminators, but at SNR

---

[2]We would like to thank Michael S. Braasch and Maarten U. de Haag of University of Ohio for providing the real GPS data.
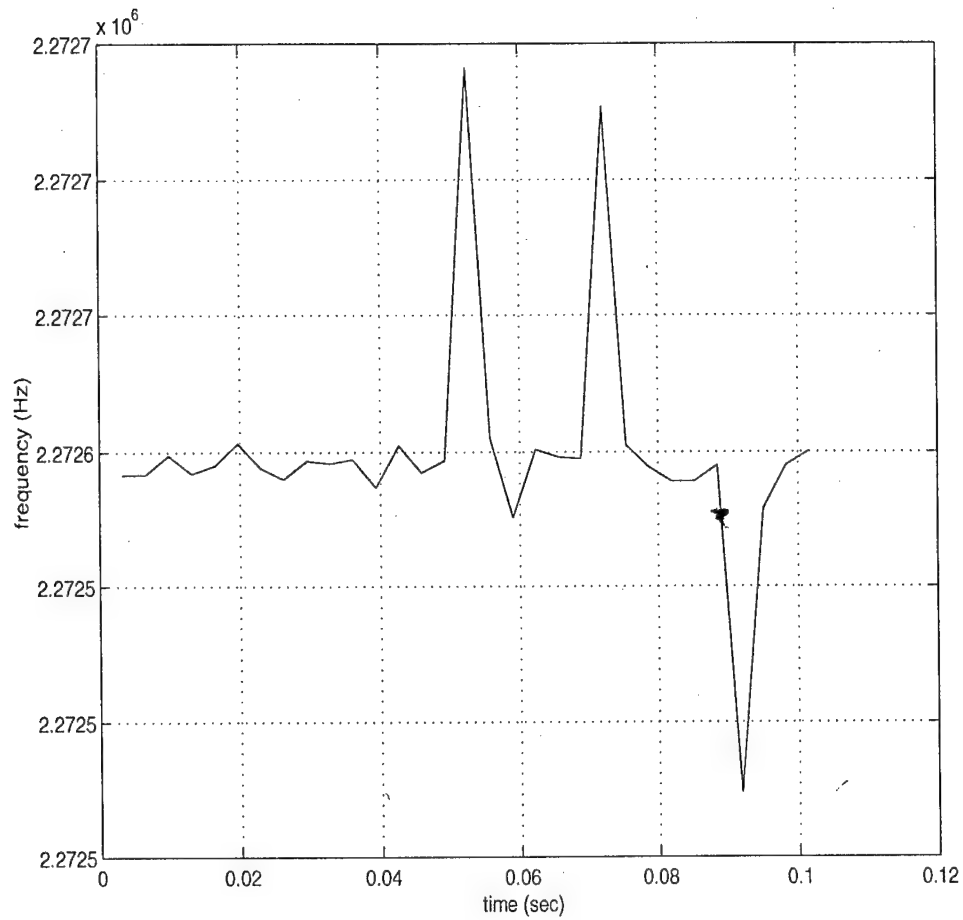
Figure 2.7: The regular FLL discriminator's frequency tracking

Table 2.1:

SNR		20	15	10	5	0
regular	lock	50	50	50	46	6
discriminator	miss	0	0	0	4	44
robust	lock	50	50	49	49	23
discriminator	miss	0	0	1	1	27

12
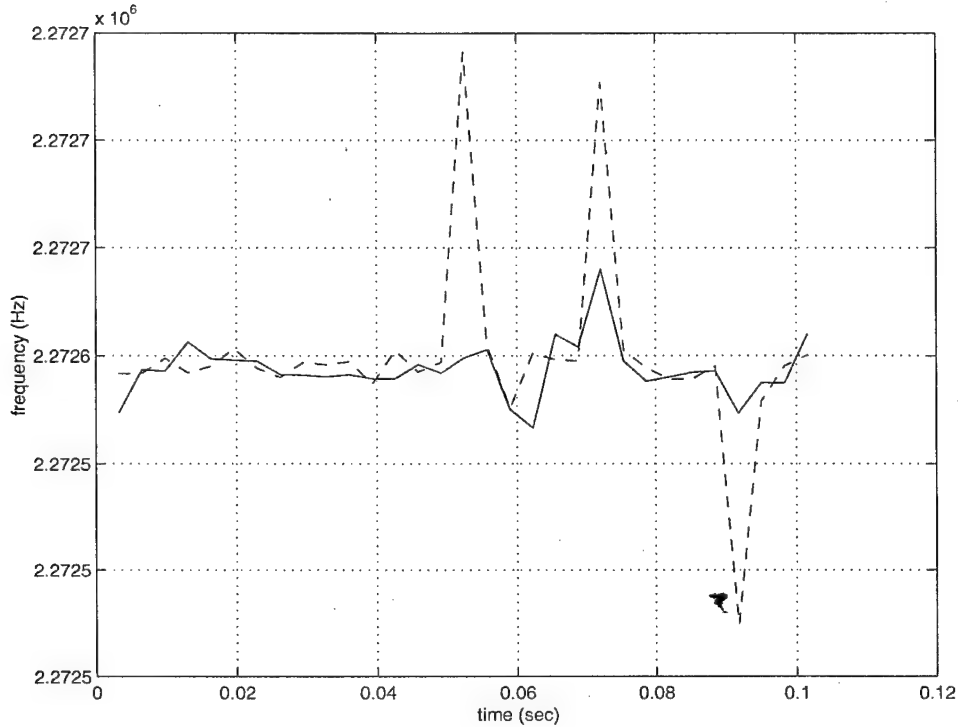
Figure 2.8: The robust FLL discriminator's frequency tracking

levels where the regular discriminator looses the signal, the robust discriminator still offers accurate signal tracking.

2.6   CONCLUSION

So far, we have presented a performance analysis on the proposed frequency lock and code delay lock discriminators on real life GPS data, and compared their performances to two existing discriminators. The proposed frequency lock discriminator proved to be robust to navigation data bit transitions. This is a big problem for regular discriminators as it limits the predetection integration time and the noise performance thereafter. The use of a robust frequency lock discriminator therefore should alleviate the 20 msec restriction on the predetection integration time, making operation possible in much heavier attenuation, noise or interference environments.

In comparison to envelope detection, the proposed code delay lock discriminator showed better performance in establishing a delay lock under real noise conditions This confirms the simulated results presented in [16], and offers promising functionality of the newly proposed techniques in adverse signal conditions.

13

Part II

Nonlinear Diffusion:A Probabilistic Approach

Chapter 3

Anisotropic Diffusion

3.1   Introduction

Scale-based analysis has played an increasingly important role in signal and image analysis since Witkin's ground breaking paper[17], in which a so-called linear scale space was constructed. This was based on the conclusion that convolving a signal with a Gaussian kernel was equivalent to evolving it with a Heat differential operator where time plays the role of scale [18, 19]. This linear filtering approach, however, presented a major limitation, namely that important details in signal/image also get smoothed away along with the noise. On the other hand, and with a different twist, Mallat [20] proposed a systematic nonlinear multi-scale analysis framework using wavelet bases. The computational efficiency of a wavelet analysis together with its ability to focus on and localize salient features of a signal via its multi-scale coefficients have secured it great popularity. This framework provided a solid foundation for a number of subsequent studies on De-noising, segmentation, etc. [21, 22]. The implicit/explicit assumption of independence of the wavelet coefficients in the analyses of these De-noising techniques and others turned out to be a limitation in some cases and degraded their performance and overall effectiveness in others. An attempt to mitigate such a problem was provided for 1-dimensional signals in [23] by accounting for some Markovian structure, the solution appeared to improve performance albeit at a substantial computational cost. Its applicability for images, to the authors' best knowledge, was never given. Given the intrinsic ability of the continuous scale approach to preserve intra-scale correlation information as well as inter-scale information, it is reasonable to expect techniques based on such approach to result in more efficient and effective filtering. Such a clever notion was first used by Perona and Malik in their landmark paper [24] where they aimed at preserving important sharp features such as edges in images. Their technique may also be viewed as a nonlinear filter whose selective smoothing is based upon the computed local gradient (maximal smoothing in low gradient or homogeneous regions, and minimal smoothing in high gradient regions). The novelty of this approach together with its most promising results triggered a tremendous

research activity in computer vision and applied mathematics (see [25] for a good review of the literature as well some other fundamental papers such as [26, 27]).

A number of very good papers have recently provided inspiring variational interpretations to various nonlinear smoothing techniques [25]. Specifically, many existing nonlinear evolution equations were shown to result from a minimization of energy functionals whose Euler-Lagrange equations led to a gradient descent method, thereby giving rise to a PDE. All of these evolution techniques required an a priori stopping time, the absence of which almost always led to, as is well known, to a complete smoothing of the signal/image (i.e., the steady state of the PDE). Furthermore most of the existing developments, if not all, have been predominantly deterministic in nature, with little to no stochastic treatment or interpretation of the diffusion. The characteristics of the random process which underlies a diffusion have hence been overlooked, and their overall influence on the solution in different scenarios has remained unclear. Pollak *et. al.* [5] recently proposed an approach addressing robustness issues, and showed some remarkable results for a wide class of perturbation noises. The analysis remained as in all other cases, fundamentally deterministic, and also required knowledge of the stopping time for the evolution.

Towards understanding the intrinsic stochastic behavior of nonlinear diffusions, we adopt a probabilistic view of diffusion (and partially described) in [11], and propose a framework where nonlinear diffusions are cast and are given an insightful interpretation. Additional extensions are also proposed in [12]. In this report we describe in sufficient detail an alternative view of nonlinear filtering/diffusion which may also be found in [6]. This in fact is instrumental in our providing an alternative interpretation of existing methods, e.g., Perona-Malik equation, and in using the gained insight to propose a solution to its well known limitations. More specifically, we view an evolution equation as a solution to a controlled diffusion [28] resulting from an optimization of an energy functional. This ultimately leads to a two to four state Markov Chain (MC) with one step transition probabilities well adapted to preserving the salient features of an image/signal (such as edges) while smoothing away the noise. As will be elaborated on further below, in addition to a marked performance improvement over P-M equation, and by way of our newly proposed technique we are able to lift a *longstanding problem in nonlinear diffusion*, namely requiring prior knowledge of a stopping time. We in fact show that the stable point for our equation is a staircase function. The immediate applications of such a filtering technique are signal/image enhancement as well as segmentation and feature extraction [7].

Some background material is first presented in the next section and prior to our reformulation of the diffusion problem in Section 3. Following a discrete formulation, we proceed to alternatively reinterpret in Section 4 nonlinear diffusion equations as non-homogeneous controlled Markov Chains. In Section 5, we use the insight gained in the previous sections to propose a new algorithm whose evaluation through numerous substantiating examples is provided in Section 6. We finally give some concluding remarks in Section 7.

16

3.2   Probabilistic View of Diffusion

A stochastic process X_t may be defined as a parameterized collection of random variables $\{X_t\}_{t \in T}$ defined on a probability space $(\Omega, \mathcal{F}, P)$ and assuming values in $\mathcal{R}^n$ in general, so that,

$$\forall \omega \in \Omega, \, \omega \rightarrow X_t(\omega), \, t \in T,$$

where Ω is the usual sample space, $\mathcal{F}$ the σ-field and P the probability measure. A nice and intuitively appealing interpretation for ω is that of a particle whose position at time t is given by $X_t(\omega)$.

Definition 1. *A stochastic process which satisfies the following stochastic differential equation(SDE) is called an Ito-diffusion[29].*

$$dX_t = b(t, X_t)dt + \sigma(t, X_t)dB_t, \tag{3.1}$$

where B_t is a standard Brownian motion and $b(t, x), a(t, x)$ are drift and diffusion coefficients, $(d = 1, 2$ in this paper). Furthermore, if the drift and diffusion terms $b(t, x), a(t, x)$ are associated with a function $v(t, x)$, $X_t^v = X_t$ is called a controlled diffusion, where X_t^v satisfies the SDE

$$dX_t^v = b(t, v(t, X_t))dt + \sigma(t, v(t, X_t))dB_t. \tag{3.2}$$

Denote by $p(s, x, t, dy)$ the probability transition function of a stochastic process X_t, i.e., $p(s, x, t, dy) = P(X_t \in dy | X_s = x)$, and by $p(s, x, t, y)$ the probability transition density function of a particle starting at location x and time s and reaching y at time t. The transition probability is normally determined by the drift and diffusion coefficients, which characterize how the diffusion behaves. If there is a control factor in the coefficients, it is reflected by a constrained transition probability for the particle. The infinitesimal generator (i.e., continuous operator which described such a motion) of the diffusion in Eq. (3.2) can be then written as :

$$L_t^v = \frac{1}{2} \sum_{i,j=1}^{d} a^{ij}(t, v(t, x)) \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i,j=1}^{d} b^i(t, v(t, x)) \frac{\partial}{\partial x_i}$$

where $\mathbf{a}(t, v(t, x)) = \sigma(t, v(t, x))^T \sigma(t, v(t, x))$, The solution of the resulting PDE

$$\begin{aligned} \frac{\partial U_t(x)}{\partial t} &= L_t^v U_t(x) \\ U(s, x) &= f(x) \quad \text{for some} \quad s \geq 0, \end{aligned} \tag{3.3}$$

can be written as an expected value $U_t(x) = E_x\{f(X_t)\} = \int f(y) P(s, x, t, dy)$ [30]. If we let $s = 0$., Eq.(3.3) is the so-called Markov forward equation. A backward equation can be viewed as a reverse time process, with the following form:

Figure 3.1: A particle (pixel) may diffuse over many possible paths, and an average is usually computed.

$$\frac{\partial U_t(x)}{\partial t} + L_t^{v*} U_t(x) = 0$$

$$U_T(x) = f(x) \quad \text{for some} \quad T \geq 0, \tag{3.4}$$

where $L_t^{v*}(\cdot)$ is an adjoint operator of L_t^v. The solution can again be expressed as $U_t(x) = E_{t,x}(f(X_T))$, where T is the fixed terminal/end time of the diffusion (or initial in the case of an inverse diffusion). Note that a backward diffusion equation can be viewed as a forward diffusion by merely selecting $t' = T - t$.

An illustrating example of a non-controlled diffusion is the process described by the PDE in Eq.(3.8), in which L_t^v is specified as a Laplacian operator Δ (i.e., $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$). Diffusion of heat in a homogeneous medium fundamentally stems from the motion of particles. It can be shown that the inherent randomness of this motion is well-described by a Brownian motion B_t [29], where an individual outcome $\omega \in \Omega$ in the prevailing sample space, may be associated to a particle. The process B_t can then be interpreted as the distance traveled by particle ω at time t. It is well known that such a transition density for a Brownian motion in 1-D case, for instance, is a Gaussian PDF $p(t, x, y) = \frac{1}{(2\pi t)^{1/2}} e^{-\frac{(y-x)^2}{2t}}$ $\forall\, x, y \in \mathcal{R}, t > 0$ (recall that a Brownian motion has independent Gaussian increments). It is thus clear that a stochastic interpretation of a solution (if it exists) subject to some differentiability conditions, can be given by way of an ensemble average [30]

$$U_t(x) = \mathrm{E}_x\{f(B_t)\} = \int_{\mathbb{R}} p(t, x, y) f(y) dy, \tag{3.5}$$

where the expectation $\mathrm{E}(\cdot)$ is computed over all possible reachable positions x_1 starting at position x. In the 2-D case, it is similarly possible to have such an interpretation as displayed in Fig. 3.1. The times $t = t_1$ and $t = t_2$ are the instants at which all possible positions are averaged to yield a solution at the respective times.

3.3 Diffusion on a Lattice

As previously noted, our chief interest here is to propose a framework within which a stochastic interpretation of a diffusion (or more generally of the so-called scale space analysis) is achieved, and is in turn instrumental in gaining insight. Towards that end and to further extend and possibly improve on existing techniques, we begin by discretizing the space as well as the scale/time variables.

18

3.3.1 Discrete Approximation of Diffusion

Recall that a symmetric one-dimensional (1-D) random walk is well known to converge to a Brownian motion as $\tau \to 0$ and $\delta \to 0$, with τ, δ respectively denoting scale and distance discrete steps. A particle following such a trajectory will move on a 1-D lattice with probability $1/2$ to the left or to the right, while on a 2-D plane, it will move to any of the four nearest neighbors (east, west, north, south) with equal probability of $1/4$. Formally, in 2-D space, we write the spatial variable $(x_1^i, x_2^i) = (x_1 + i\delta, x_2 + i\delta)$ with $i \in \mathbb{Z}$ and the scale $t_n = n\tau$ with $n \in \mathbb{N}$, and we denote the one step transition probability of a particle from (x_1^0, x_2^0) to (x_1, x_2) at the nth scale step, by $p_n((x_1^0, x_2^0), (x_1, x_2))$. As a result, we obtain a standard form from Eq. (3.5), namely the probability of a particle being at (x_1, x_2) at scale/time $(n+1)^{st}$ step as $\tau \to 0$ and $\delta \to 0$,

Proposition 1. *The following discrete equation,*

$$
\begin{aligned}
p_{n+1}((x_1^0, x_2^0), (x_1, x_2)) &= \tfrac{1}{4}p_n((x_1^0, x_2^0), (x - \delta, y) + \tfrac{1}{4}p_n((x_1^0, x_2^0), (x_1 + \delta, x_2)) \\
&\quad + \tfrac{1}{4}p_n((x_1^0, x_2^0), (x_1, x_2 - \delta)) + \tfrac{1}{4}p_n((x_1^0, x_2^0), (x_1, x_2 + \delta))
\end{aligned}
\tag{3.6}
$$

converges to

$$
\frac{\partial p_t((x_1^0, x_2^0), (x_1, x_2))}{\partial t} = \frac{\partial^2 p_t((x_1^0, x_2^0), (x_1, x_2))}{\partial x_1^2} + \frac{\partial^2 p_t((x_1^0, x_2^0), (x_1, x_2))}{\partial x_2^2}
\tag{3.7}
$$

Proof: Subtracting $p_n(x_1^0, x_2^0)$ from both sides of Eq. (3.6) and dividing it by τ, we obtain

$$
[p_{n+1}((x_1^0, x_2^0), (x_1, x_2)) - p_n((x_1^0, x_2^0), (x_1, x_2))]/\tau =
$$
$$
\tfrac{1}{4\tau}[p_n((x_1^0, x_2^0), (x_1 - \delta, x_2)) - 2p_n((x_1^0, x_2^0), (x_1, x_2)) + p_n((x_1^0, x_2^0), (x_1 + \delta, x_2))] +
$$
$$
\tfrac{1}{4\tau}[p_n((x_1^0, x_2^0), (x_1, x_2 - \delta)) - 2p_n((x_1^0, x_2^0), (x_1, x_2)) +
$$
$$
p_n((x_1^0, x_2^0), (x_1, x_2 + \delta))]
$$

which upon letting $\tau = \delta^2/4$ and $\delta \to 0$, concludes the proof. ∎

The numerical approximation of a linear diffusion is hence readily implemented by way of a random walk.

When considering a 1-D Brownian motion on a compact interval, a reflecting wall should be accounted for, making the Markov Chain (MC) aperiodic and recurrent and for which a solution to $\Delta U(\vec{x}) = 0, U_0(\vec{x}) = f(\vec{x})$ takes the form $U(\vec{x}) = E_{\vec{x}}(f(X_T))$. This also implies a discrete solution $U(\vec{x}) = \sum_i p_i f(X_i)$ where p_i is the probability of a particle to be in state i as t grows large. The independence of the solution $U(\vec{x})$ of the initial state, is equivalent to its convergence to some mean value of $f(\vec{x})$ as $\vec{x}$ is averaged over all possible

19

paths. This in a sense provides an intuitive justification for the convergence of a heat equation to a constant. The case in point arises when we are faced with a deterministic diffusion X_t, which is alternatively expressed as $dX_t = [10]dt$. The corresponding solution obtained from Eq. (3.3) is $U(\vec{x}) = f(x_{01} + t, x_{02})$, where $\vec{x_0}$ is the initial state, clearly non-constant as expected.

In light of the foregoing development, and for better insight and intuitive clarity, we find it useful to carry out most of the exposition and the analysis in a discrete setting. Prior to delving into our formulation and interpretation of a Non-Linear (NL) diffusion, we present an illustrative example where a so-called controlled diffusion leads to a Markov Chain as a result of an optimization.

3.3.2 Finite Markov Chain Example

We start with a finite Markov chain [31] as in Fig. 3.2 with three possible states α, β, γ with respective costs 1, 2, 3. Our goal is to select a "best" strategy (minimum cost) for a particle to make a two-step transition from a state, say α. The corresponding transition probabilities are obtained from the following sets of strategies:

$$
\begin{aligned}
G_\alpha &= \{(\frac{1}{2}, 0, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}, 0)\} \\
G_\beta &= \{(\frac{1}{3}, 0, \frac{2}{3}), (\frac{3}{4}, \frac{1}{4}, 0)\} \\
G_\gamma &= \{(1, 0, 0), (\frac{1}{2}, 0, \frac{1}{2}), (\frac{1}{3}, 0, \frac{2}{3})\}.
\end{aligned}
$$

For state α for instance, two choices are possible:

- the first strategy has it move to state γ with probability $\frac{1}{2}$, and remain stationary with probability $\frac{1}{2}$,

- the second lets it move to state β with probability $\frac{1}{2}$ and remain stationary with probability $\frac{1}{2}$.

Taking into account the respective costs as well as the transition probabilities, an optimal strategy for α is determined to be $(\frac{1}{2}, \frac{1}{2}, 0)$ with a minimum cost of $\frac{3}{2}$, while in state β a cost of $\frac{5}{4}$ with the strategy $(\frac{3}{4}, \frac{1}{4}, 0)$, and in state γ we obtain a cost of 1 with strategy $(1, 0, 0)$. A second step transition may be similarly found, with respective costs for α, β, γ, of $\frac{5}{4}, \frac{7}{6}, \frac{7}{6}$ resulting from $(\frac{1}{2}, 0, \frac{1}{2}), (\frac{1}{3}, 0, \frac{2}{3}), (\frac{1}{3}, 0, \frac{2}{3})$ respectively. This process may be continued indefinitely.

Note that more complex strategies are possible and may be constructed for the intermediate steps, e.g., variable strategies along the steps, etc.. When, on the other hand, a given strategy set only depends on the previous state, it is referred to as a Markov strategy. It is also clear from the foregoing example that the resulting process, by way of its transition

In an attempt to solve Eq.(3.8) and using its stochastic interpretation, we proceed to write

$$U_t(\vec{x}) = E_{t,\vec{x}}(f(X_T)),$$ (3.10)

where $f(\cdot) = U(0, \vec{x})$ is the initial data. In particular, we obtain one step transition as

$$
\begin{aligned}
U_t(x) &= E_{t,x}(E_{t-\tau}(f(X_T))) \\
&= \int E_{t-\tau,y}(f(X_T))p(\tau, x, y)dy \\
&= \int U_{t-\tau}(y)P_\tau(y|x)dy.
\end{aligned}
$$ (3.11)

Note that the above diffusion is a backward diffusion which, as mentioned in Section 2, is treated as a forward diffusion for the sake of clarity. The probability $P_\tau(y|x)$ should then be interpreted as $P(\xi_{t-\tau} = \vec{y}|\xi_T = \vec{x})$ to emphasize the backward evolution in time/scale.

3.4.1   Discrete Time/Scale Evolution

In discretizing $\vec{x}$ and t, we account for a reverse time evolution by relabeling time "$t - \tau$" by 1 and "$t - n\tau$" by n, hence making a backward diffusion equation look more like a forward diffusion. We denote by $U_n(\vec{x})$ the value of the solution at time step $n\tau$ and location/state $\vec{x}$. Eq.(3.11) can then be written in the form,

$$U_{n+1}(x) = \sum_{\vec{y}} U_n(y)P_n(y|x).$$ (3.12)

Since the solution to Eq.(3.7) is a Gaussian transition density function, it characterizes the evolution of a particle along a Brownian trajectory starting at $\vec{x_0}$ and time t. Using the fact that a limiting process of a random walk is a Brownian motion, we may compute the solution to Eq. (3.12) at any desired discrete time/scale. At the first time step τ and for a 1-D case, we can write

$$U_1(x) = \frac{1}{2}f(x - \delta) + \frac{1}{2}f(x + \delta),$$ (3.13)

while in a 2-D scenario, we have

$$U_1(x_1, x_2) = \frac{1}{4}f(x_1 - \delta, x_2) + \frac{1}{4}f(x_1 + \delta, x_2) + \frac{1}{4}f(x_1, x_2 - \delta) + \frac{1}{4}f(x_1, x_2 + \delta),$$ (3.14)

both of which are the result of an averaging process. More generally, we can write the solution to the linear heat equation as a discrete expectation for respectively 1-D and 2-D as

$$U_{n+1}(x) = \frac{1}{2}U_n(x - \delta) + \frac{1}{2}U_n(x + \delta),$$ (3.15)

22

$$U_{n+1}(x_1, x_2) = \frac{1}{4}U_n(x - \delta, y) + \frac{1}{4}U_n(x_1 + \delta, x_2) + \frac{1}{4}U_n(x_1, x_2 - \delta) +$$
$$\frac{1}{4}U_n(x_1, x_2 + \delta) \tag{3.16}$$

$$\tag{3.17}$$

Due to the underlying random walker moving to its neighbor with 1/2 probability in 1-D (and to its four nearest neighbors with probability 1/4 in 2-D), it is clear that the linear evolution will indiscriminately smooth away sharp features along with the noise.

3.4.2 A Stochastic View of Perona-Malik Diffusion

The linear stochastic differential equation led to a linear diffusion by way of a Laplacian as its corresponding infinitesimal generator. Using this development as an inspiration, together with its discrete stochastic formulation and interpretation, we proceed in an analogous manner to rewrite the P-M equation to be interpreted as a particle-based diffusion.

Proposition 2. *Based on a particle system interpretation, P-M equation may rewritten as*

$$U_{n+1}(x) = p_n(x, x + \delta)U_n(x + \delta) + p_n(x, x - \delta)U_n(x - \delta),$$
$$[1 - p_n(x, x + \delta) + p_n(x, x - \delta]U_n(x). \tag{3.18}$$

Proof: The proof follows immediately from the discretization of Eq.(3.9) and rewriting $1/2g\,(|\,U_n(x \pm \delta) - U_n(x)\,|) = p_n(x, x \pm \delta) = p_n(\xi_{n+1} = x \pm \delta \mid \xi_n = x)$ to denote the transition probability of a Markov chain $\{\xi_{(\cdot)}\}$ to move from state x to state $x \pm \delta$.

A similar expression for a 2-D signal (image) may be written as

$$U_{n+1}(x_1, x_2)$$
$$= p_S^{n+1}(x_1, x_2)U_n(x_1 + \delta, x_2) + p_N^{n+1}U_n(x_1 - \delta, x_2)$$
$$+ p_E^{n+1}(x_1, x_2)U_n(x_1, x_2 + \delta) + p_W^{n+1}U_n(x_1, x_2 - \delta)$$
$$+ \ [1 - p_S^{n+1}(x_1, x_2) - p_N^{n+1}(x_1, x_2) - p_E^{n+1}(x_1, x_2) - p_W^{n+1}(x_1, x_2)]U_n(x_1, x_2),$$
$$\tag{3.19}$$

where

$$p_S^{n+1}(x_1, x_2) = \frac{1}{4}g\,(|\,U_n(x_1 + \delta, x_2) - U_n(x_1, x_2)\,|),$$

$$p_N^{n+1}(x_1, x_2) = \frac{1}{4}g\,(|\,U_n(x - \delta, y) - U_n(x_1, x_2)\,|),$$

$$p_E^{n+1}(x_1, x_2) = \frac{1}{4}g\,(|\,U_n(x_1, x_2 + \delta) - U_n(x_1, x_2)\,|),$$

$$p_W^{n+1}(x_1, x_2) = \frac{1}{4}g\,(|\,U_n(x_1, x_2 - \delta) - U_n(x_1, x_2)\,|)$$

represent the transition probabilities of the underlying Markov chain ξ_n, i.e.,

$p_S(\xi_{n+1} = (x_1 + \delta, x_2) \mid \xi_n = (x_1, x_2)) = p_S^{n+1}(x_1, x_2)$. These equations are intuitively appealing, in that the random walk of a particle/pixel (or the diffusion) ξ_n takes place according to the prevailing one sided gradient at position (x_1, x_2) in any of the four directions. At time step $n + 1$, a south (resp. north, east, west) moving walk takes place with probability $p_S^{n+1}(x_1, x_2)$ (resp. $p_N^{n+1}(x_1, x_2)$, $p_E^{n+1}(x_1, x_2)$, $p_W^{n+1}(x_1, x_2)$), and the particle remains in place with probability $p_0^{n+1}(x) = 1 - p_S^{n+1}(x_1, x_2) - p_N^{n+1}(x_1, x_2) - p_E^{n+1}(x_1, x_2) - p_W^{n+1}(x_1, x_2)$. It is clear here that the transition probability of a random walk is determined by the gradient, thus, the resulting diffusion is well controlled. This is in sharp contrast to the linear diffusion where the random walk invariably takes place with a constant probability of 1/4. Note that while the derivation of an exact SDE corresponding to P-M equation as an infinitesimal generator, is interesting in and of itself, it requires a more complex system of particles which is of no additional insight and of little relevance to our stated goal in this paper.

3.5 Two Sided Gradient-Driven Diffusion

3.5.1 A variation on a theme

As discussed in Section 2, at each scale of our analysis, the mean value of the process $U(\cdot, \cdot)$ is evaluated as a result of a non-homogeneous random walk with the transition probability controlled by the underlying process at the previous scale. In addition, and to avoid potential stability problems, we ensure that the probability of a jump of a particle (pixel) farther than an immediate neighbor is zero, which effectively emulates a continuous diffusion. Furthermore, we ensure that there always be a one step transition of a particle to its neighbors to avoid a slowdown in convergence due to likely stationary states [28]. We adopt this paradigm to construct a non-homogeneous Markov chain whose transition probabilities are based on the current particle states and their functional value. This results in a set of consecutive transition steps through scales, each in a sense, defining a new random process with a new probability transition.

While the goal in signal/image processing is to maximally smooth out the noise, we are also keen on achieving a solution that is as faithful as possible to the initial underlying signal. To thus help better localize the homogeneous regions together with their boundaries, we use in our transition dynamics a bidirectional gradient-based "probability measure". (subgradient in continuous space). Using the Szökefalvi-Nagy's inequality[32], to optimize the gradient energy (to delineate regions), we have to minimize the following energy expression,

$$\mathcal{E}(U_{n+1}) = \sum_x \mathcal{E}(U_{n+1}(x))$$

$$= \sum_x [(U_n(x+\delta) - U_n(x))(U_{n+1}(x) - U_n(x-\delta))]^2$$

$$+ \quad [(U_n(x-\delta) - U_n(x))(U_{n+1}(x) - U_n(x+\delta))]^2 \qquad (3.20)$$

where $U_{n+1}(x)$, assumed to result from Eq.(3.11), is written as

$$U_{n+1}(x) = P_{n+1}(x-\delta|x)U_n(x-\delta) + P_{n+1}(x+\delta|x)U_n(x+\delta), \qquad (3.21)$$

with $P_{n+1}(x-\delta|x) + P_{n+1}(x+\delta|x) = 1$. Minimizing Eq.(3.20) entails an appropriate choice of a probability measure as follows,

Theorem 1. *The transition probability solving Eq.(3.20) is given by*
$P_{n+1}(x-\delta|x) = P\{\xi_{n+1} = x - \delta|\xi_n = x\}$ *with*

$$P_{n+1}(x-\delta|x) = \frac{|U_n(x+\delta) - U_n(x)|^2}{|U_n(x-\delta) - U_n(x)|^2 + |U_n(x+\delta) - U_n(x)|^2}, \qquad (3.22)$$

where $U_{n+1}(x)$ satisfies Eq.(3.21).

(See Appendix A for Proof). ∎

For a 2-D image, we denote the transition probability by $p_S^{n+1}(x_1, x_2) = P\{\xi_{n+1} = (x_1 + \delta, x_2)|\xi_n = (x_1, x_2)\}$ (similarly for other probabilities) and obtain the following expression for the transition probability

$$p_S^{n+1}(x_1, x_2) = \frac{\mathcal{N}}{\mathcal{S} + \mathcal{N} + \mathcal{E} + \mathcal{W}}, \qquad (3.23)$$

where

$$\begin{aligned}
\mathcal{N} &= |U_n(x-\delta, y) - U_n(x_1, x_2)|^2 \\
\mathcal{S} &= |U_n(x_1+\delta, x_2) - U_n(x_1, x_2)|^2 \\
\mathcal{E} &= |U_n(x_1, x_2+\delta) - U_n(x_1, x_2)|^2 \\
\mathcal{W} &= |U_n(x_1, x_2-\delta) - U_n(x_1, x_2)|^2
\end{aligned}$$

Using the above transition probability, our newly proposed diffusion is written as

$$\begin{aligned}
U_{n+1}(x_1, x_2) = \quad & U_n(x_1+\delta, x_2)p_S^{n+1}(x_1, x_2) + U_n(x-\delta, y)p_N^{n+1}(x_1, x_2) \\
& U_n(x_1, x_2+\delta)p_E^{n+1}(x_1, x_2) + U_n(x, y-\delta)p_W^{n+1}(x_1, x_2). \qquad (3.24)
\end{aligned}$$

challenges to simple gradient-based segmentation and/or linear filtering, and this is for the most part due to their impulsive nature. The results of such approaches usually results in visually unpleasant and quantitatively inaccurate results if at all. In Fig. 3.6, the potential for complete diffusion rather quickly (for an inadequate choice of the threshold parameter in the P-M equation) is exhibited whereas and as demonstrated above, the new approach will stabilize with no parameter adjustment. For a well known stopping time and well chosen parameter, P-M approach performs quite well, this may, however, may turn out to be a limitation for a number of real applications.

In Fig. 3.7 an enhancement/De-blurring-like effect using our algorithm is also demonstrated for a checker board. The price for avoiding the explicit knowledge of the stopping time using our approach, is the arising of block effects which although common to many existing techniques is a drawback. Although this may be fine tuned away for pure De-noising purposes, we are currently looking into techniques specifically addressing such an issue. In Figure 3.7, a De-blurring example is shown, demonstrating the capacity of the algorithm to enhance edges and again stabilize at staircase functions.

For establishing a more quantitative measure of performance we use the figures in Figure 3.8. A pixel deviation is computed and a Monte Carlo simulation is conducted to construct an error rate curve which is consistent with our visual assessment and displayed in Fig. 3.9.

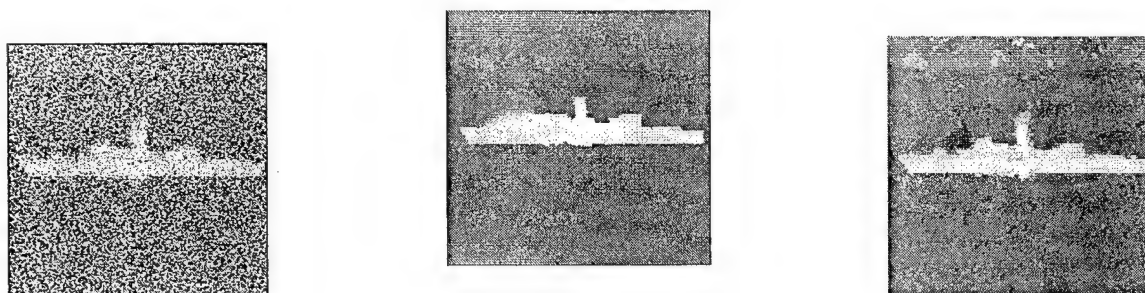Figure 3.3: Stable signal remains unchanged following proposed nonlinear diffusion.

Figure 3.4: A noisy image together with its enhanced copy by the proposed algorithm and by the P-M method best result.
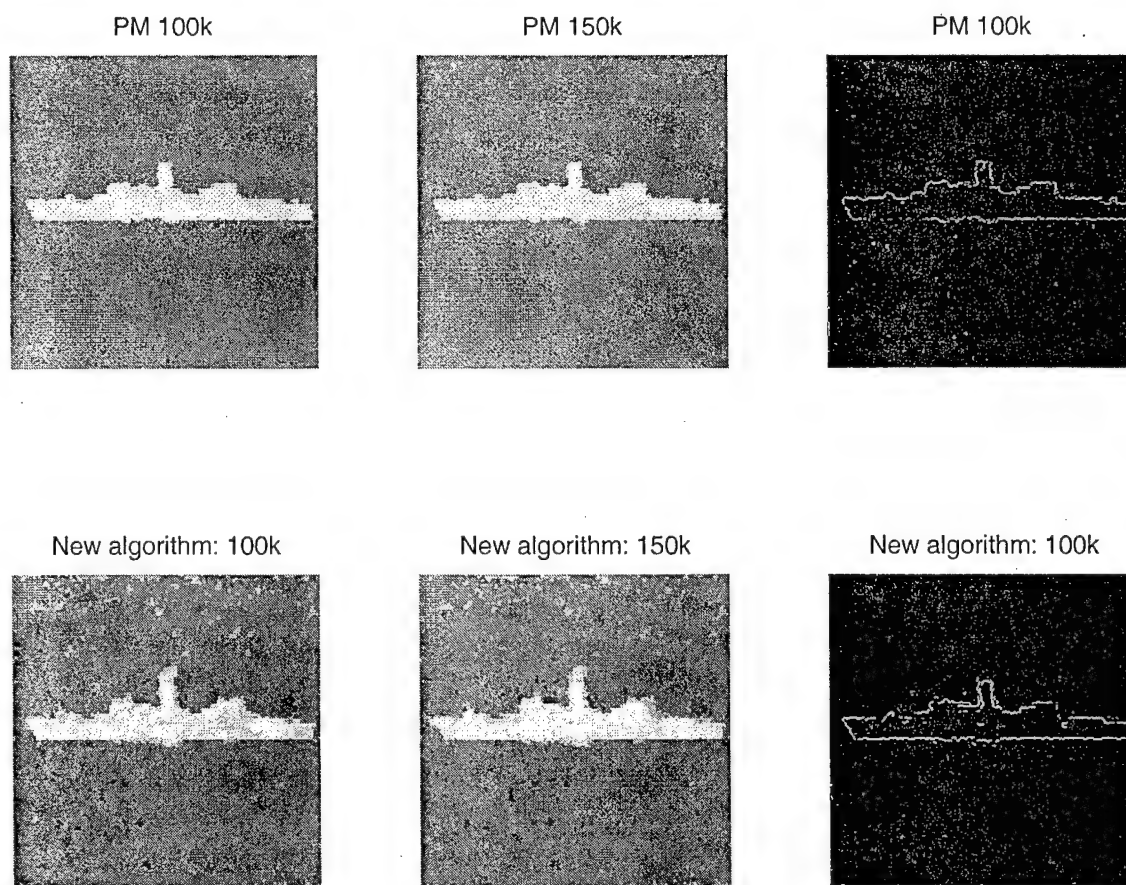

Figure 3.5: Complete Smoothing vs Stability.

PM 100   PM 200   PM 1000

PM 5000   PM 8000   PM 10k

Figure 3.6: PM algorithm.

Sinusoid Image   Filtered Sinusoid Image
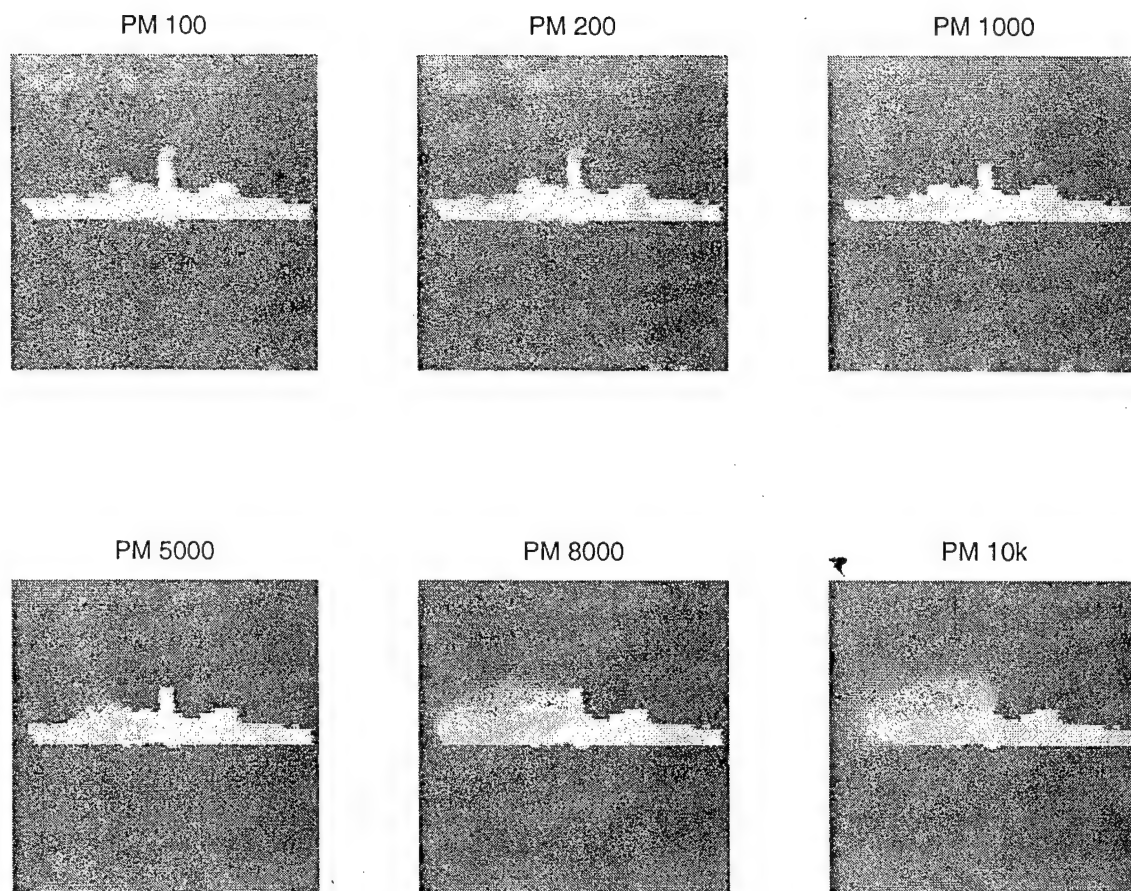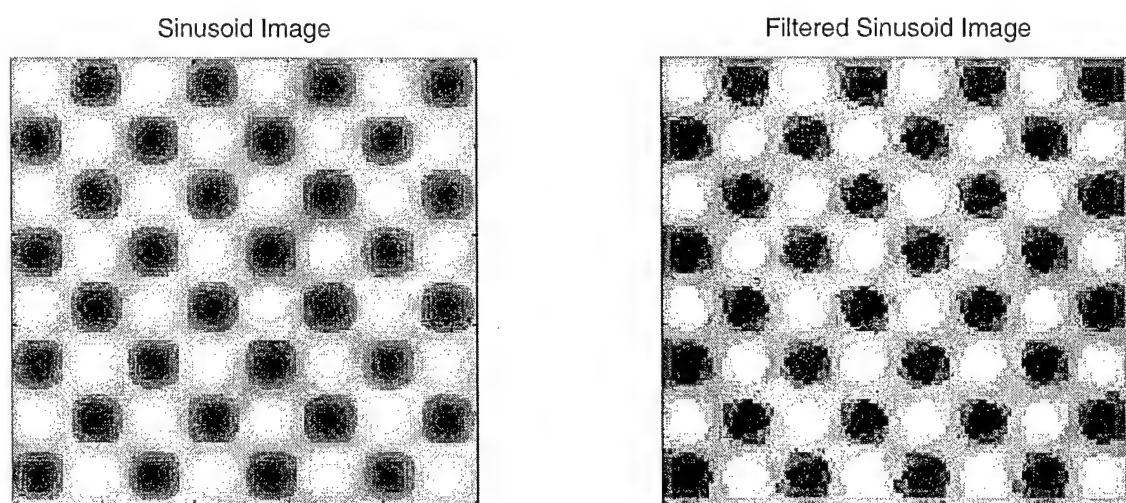
Figure 3.7: Checker Board Enhancement.

3.7   Conclusion

The proposed stochastic interpretation together with its link to controlled diffusion are shown to not only explicate existing techniques and their limitations, but to also provide sufficient insight to develop other novel physically and geometrically driven methodologies. We have also succeeded in resolving in part a well known and long standing problem of unknown stopping criterion.

Chapter 4

Geometric Stochastic Flows

4.1 Introduction

Closely related techniques to the above anistropic diffusions are those of curve evolutions. As we elaborate further in the next section, the so-called geometric heat flow may be viewed as a descendent of the linear heat flow, in spite of the fact that the general class of curve evolutions itself need not be so. Curve evolution has, in recent years, emerged as an important application of partial differential equations (PDE's) in image processing, computer vision, and computer graphics. Their applications to individual curves such as in edge-detection, skeletonization, and shape analysis, but have also been extended to their simultaneous action on the level sets of an image in a number of geometrically based anisotropic smoothing algorithms. A detailed account of these methods may be found in [33, 34] and also in, e.g. [7].

If run for too long, however, even large scale features will be destroyed. The reason stems from the fact that as the geometric heat flow shrinks any closed curve, the curve becomes more and more circular (elliptical in the case of the affine flow) and will eventually collapse into a single point [35]. It is therefore not always possible to preserve desired features in the shapes of objects (corners for example) if too much evolution is required to remove a significant level of noise. Furthermore, it is not well understood how these curvature-based filters are affected by different noise distributions and when this sort of problem may occur. Much like the anisotropic diffusion problem, the curve evolution technique has largely been addressed in the literature, outside [2, 36], in a purely deterministic setting. In this part of our research effort, we provide a stochastic formulation of the geometric heat equation and use the resulting insights to develop a new class of curvature-based flows and anisotropic diffusion filters which preserve desired features in the shape of an object. Under these new flows, evolving curves take the limiting form of a polygon (see [37] for evolutions of polygons related to the geometric and affine geometric heat flows). The resulting diffusion models may therefore be applied for much longer periods of time without distorting the shapes of polygonal objects in the image, thereby mitigating the

We may write these directions in terms of the first derivatives of the image as

$$\eta = \frac{(u_x, u_y)}{\sqrt{u_x{}^2 + u_y{}^2}}, \quad \xi = \frac{(-u_y, u_x)}{\sqrt{u_x{}^2 + u_y{}^2}},$$

Since these constitute orthogonal directions, we may exploit the rotational invariance of the Laplacian operator and re-write the linear heat equation in terms of these two variables:

$$u_t = \nabla \cdot (\nabla u) = u_{\xi\xi} + u_{\eta\eta}$$

where $u_{\eta\eta}$ and $u_{\xi\xi}$ denote second-order directional derivatives in the directions of η and ξ respectively. It is possible to derive the following expressions

$$u_{\eta\eta} = \frac{u_x^2 u_{xx} + 2u_x u_y u_{xy} + u_y^2 u_{yy}}{u_x^2 + u_y^2} \tag{4.2}$$

$$u_{\xi\xi} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2} \tag{4.3}$$

By subtracting the normal diffusion component (4.2) from the linear heat equation, which diffuses isotropically, we obtain the following anisotropic model, which diffuses along the boundaries of image features but not across them

$$u_t = u_{\xi\xi} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2}. \tag{4.4}$$

We may obtain this same equation in a completely different and much more geometric manner by specifying the evolution of each level curve in the image. Let $\mathcal{C}$ denote a particular iso-intensity contour which we will deform over time via the following flow,

$$\mathcal{C}_t = \mathcal{C}_{ss} = \kappa \mathbf{N} \tag{4.5}$$

where s denotes the arclength parameter, κ the Euclidean curvature, and $\mathbf{N}$ the inward unit normal. Equation (4.5), referred to as the *Geometric Heat Equation* (GHE), is well known for its smoothing properties. It has been shown by Grayson [35] that any closed, embedded curve evolving according to (4.5) will convexify and smoothly shrink to a single point in finite time, becoming more and more circular along the way. This flow is also referred to as the *curve shortening flow* since it corresponds to the gradient (descent) evolution of the arclength functional. See [39, 40, 41] for a more extensive discussion of the many properties associated with this flow. Because the evolution speed is a function of the curvature at each point on a curve, this flow gives rise to a *Euclidean invariant* scale space (see [42, 43, 38]) in which finer features are removed first, followed by coarser features, as the curve evolves. A related flow, based upon the affine geometry of the curve, is given by $\mathcal{C}_t = \kappa^{1/3}\mathbf{N}$ and shares many of the same properties as the curve shortening flow but gives rise a more general *affine invariant* scale space (see [42, 44, 45]).

34

$(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x}))/\sqrt{u_x(t, \boldsymbol{x})^2 + u_y(t, \boldsymbol{x})^2}$. It follows, $\theta(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x})) = \tan^{-1}(\frac{u_y(t, \boldsymbol{x})}{u_x(t, \boldsymbol{x})})$. Using these equations, and defining an operator $\mathcal{A}_{\text{GHE}}$ of the form

$$
\begin{aligned}
\mathcal{A}_{\text{GHE}} u(t, \boldsymbol{x}) \;=\; & \sin^2 \theta(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x})) \; u_{xx}(t, \boldsymbol{x}) \\
& - 2 \sin \theta(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x})) \; \cos \theta(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x})) \; u_{xy}(t, \boldsymbol{x}) \\
& + \cos^2 \theta(u_x(t, \boldsymbol{x}), u_y(t, \boldsymbol{x})) \; u_{yy}(t, \boldsymbol{x}),
\end{aligned}
\tag{4.7}
$$

the geometric heat equation (4.4) can be re-written as

$$
\begin{aligned}
u_t(t, \boldsymbol{x}) &= \mathcal{A}_{\text{GHE}} u(t, \boldsymbol{x}), & (4.8) \\
u(0, \boldsymbol{x}) &= u_o(\boldsymbol{x}), & (4.9)
\end{aligned}
$$

where $u_o(\boldsymbol{x})$ is the initial level set function. We next show how an evolution equation in fact corresponds to an infinitesimal generator of a Stochastic Differential Equation (SDE), by using Ito diffusions and the Kolmogorov backward diffusion theorem [46, 2, 36].

4.3.1 Ito Diffusion

The diffusion of a particle is usually well modeled by an SDE which, in turn, represents the underlying microscopic process of an evolution of a pixel or a point. The dynamics of this evolution at a macroscopic level are captured by a PDE, henceforth also referred to as a generator (infinitesimal) of the diffusion.

Definition 2. *Suppose we want to describe the motion of a small particle suspended in a moving liquid, subject to random molecular bombardments. If $\boldsymbol{b}(t, \boldsymbol{x}) \in \mathbb{R}^n$ is the velocity of the fluid at a point $\boldsymbol{x} \in \mathbb{R}^n$ and time $t \in \mathbb{R}^+$, then a widely used mathematical model for the position $\boldsymbol{X}(t)$ of the particle at time t is an SDE of the form*

$$
d\boldsymbol{X}(t) = \boldsymbol{b}(t, \boldsymbol{X}(t))dt + \boldsymbol{\sigma}(t, \boldsymbol{X}(t))d\boldsymbol{B}(t), \tag{4.10}
$$

where $\boldsymbol{X}(t)$ is an n-dimensional stochastic process, $\boldsymbol{\sigma}(t, \boldsymbol{x}) \in \mathbb{R}^{n \times m}$, and $\boldsymbol{B}(t)$ is an m-dimensional Brownian motion. $\boldsymbol{b}(\cdot, \cdot)$ is called the drift coefficient, *and $\boldsymbol{\sigma}(\cdot, \cdot)$ is called the* diffusion coefficient.

The first term in this equation corresponds to a non-random/deterministic motion, whereas the second term models randomness or noise in the motion.

The solution of such an SDE may be thought of as a mathematical description of the motion of a small particle in a moving fluid, and such stochastic processes are called (Ito) diffusions [46]. For many applications, a second order partial differential operator $\mathcal{A}$ can be associated to an Ito diffusion $\boldsymbol{X}(t)$ given by Eq. (4.10). The basic connection between $\mathcal{A}$ and $\boldsymbol{X}(t)$ is that $\mathcal{A}$ is the generator of the process $\boldsymbol{X}(t)$. If $w(\boldsymbol{x}) \in C_0^2(\mathbb{R}^n)$, (i.e., it is

continuous with continuous derivatives up to order 2, and has a compact support), then $\mathcal{A}$ is given in the form

$$\mathcal{A}\, w = \frac{1}{2}\sum_{i,j}(\boldsymbol{\sigma}\,\boldsymbol{\sigma}^{\,T})_{i,j}(\boldsymbol{x})\frac{\partial^2 w}{\partial x_i\,\partial x_j} + \sum_i \boldsymbol{b}_i(\boldsymbol{x})\frac{\partial w}{\partial x_i}. \tag{4.11}$$

Next, we state a theorem, the so-called Kolmogorov's backward equation [46], which gives a probabilistic solution to linear PDE's.

Theorem 2. *Define* $\gamma(t,\boldsymbol{x}) = E^{\boldsymbol{x}}[f(\boldsymbol{X}(t))]$, *where* $\boldsymbol{X}(t) = (X^{(1)}(t), X^{(2)}(t))$, *and* $E^{\boldsymbol{x}}[\cdot]$ *is the expectation operator with respect to the probability law of* $\boldsymbol{X}(t)$ *starting at the point* $\boldsymbol{x}$, *then there exists an operator* $\mathcal{A}$ *such that,*

$$\frac{\partial\gamma}{\partial t} = \mathcal{A}\gamma, \quad t > 0,\ \boldsymbol{x} \in \mathbb{R}^2, \tag{4.12}$$

$$\gamma(0,\boldsymbol{x}) = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{R}^2. \tag{4.13}$$

4.3.2 Stochastic Formulation

In light of the foregoing development, a natural question which arises is: *given a PDE which governs a curve shortening flow, can we obtain a corresponding SDE associated with the underlying diffusion?*

The nonlinearity of GHE presents a significant challenge to find a global Ito diffusion which explains the overall microscopical behavior of the system. Our approach here for solving such a nonlinear problem is, to explore the short-time behavior by linearizing around a known (nominal) solution. The perturbation equations so obtained will be linear and hence an approximate solution to the nonlinear problem can be obtained as the nominal value plus the perturbation term. Let us denote by $u^n(t,\boldsymbol{x})$ the solution to Eq (4.8):

$$\frac{\partial u^n}{\partial t} = \sin^2\left(\tan^{-1}(\frac{u^n_y}{u^n_x})\right)\,u^n_{xx} - \sin\left(2\tan^{-1}(\frac{u^n_y}{u^n_x})\right)\,u^n_{xy} + \cos^2\left(\tan^{-1}(\frac{u^n_y}{u^n_x})\right)\,u^n_{yy},$$

and if we write $u(t,\boldsymbol{x})$ as

$$u(t,\boldsymbol{x}) = u^n(t,\boldsymbol{x}) + \epsilon u(t,\boldsymbol{x}),$$

and define the corresponding nominal angle $\theta^n(t,\boldsymbol{x}) = \tan^{-1}(\frac{u^n_y(t,\boldsymbol{x})}{u^n_x(t,\boldsymbol{x})})$, we get a linearized version of the geometric heat equation around a nominal value:

$$\frac{\partial u(t,\boldsymbol{x})}{\partial t} \approx \mathcal{A}_{\text{GHElin}}u(t,\boldsymbol{x}) \qquad = \sin^2(\theta^n(\boldsymbol{x}))\,u_{xx}(t,\boldsymbol{x}) - \sin(2\theta^n(\boldsymbol{x}))\,u_{xy}(t,\boldsymbol{x}) +$$

$$\cos^2(\theta^n(\boldsymbol{x}))\,u_{yy}(t,\boldsymbol{x}) + c(\boldsymbol{x})(-u^n_y(\boldsymbol{x})\,u_x(t,\boldsymbol{x}) + u^n_x(\boldsymbol{x})\,u_y(t,\boldsymbol{x})), \tag{4.14}$$

where $c(\boldsymbol{x}) = \frac{1}{(u^n_x(\boldsymbol{x}))^2+(u^n_y(\boldsymbol{x}))^2}\left[\sin(2\theta^n(\boldsymbol{x}))(u^n_{xx}(\boldsymbol{x}) - u^n_{yy}(\boldsymbol{x})) - \cos(2\theta^n(\boldsymbol{x}))2u^n_{xy}(\boldsymbol{x})\right]$ (see Appendix B A for details of this derivation).

In light of this, we can proceed to state the following:

Proposition 3. *The right hand side of the linear PDE in Eq. (4.14) is the generator of the following Ito diffusion satisfying the SDE*

$$\begin{pmatrix} d\boldsymbol{X}^{(1)}(t) \\ d\boldsymbol{X}^{(2)}(t) \end{pmatrix} = c(\boldsymbol{X}(t)) \begin{pmatrix} -u_y^n(\boldsymbol{X}(t)) \\ u_x^n(\boldsymbol{X}(t)) \end{pmatrix} dt + \sqrt{2} \begin{pmatrix} -\sin\theta^n(\boldsymbol{X}(t)) \\ \cos\theta^n(\boldsymbol{X}(t)) \end{pmatrix} dB(t),$$

$$(4.15)$$

Proof. The operator $\mathcal{A}_{\text{GHElin}}$ in Eq. (4.14) is first re-written as,

$$
\begin{aligned}
\mathcal{A}_{\text{GHElin}} &= \boldsymbol{b}^T(\boldsymbol{X}) \cdot \nabla + \frac{1}{2}\sigma(\boldsymbol{X})\sigma^T(\boldsymbol{X}) \odot \mathbf{H} \\
&= c(\boldsymbol{X}) \begin{pmatrix} -u_y^n(\boldsymbol{X}) \\ u_x^n(\boldsymbol{X}) \end{pmatrix}^T \cdot \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} + \\
&\qquad \begin{pmatrix} \sin^2\theta^n(\boldsymbol{X}) & -\sin\theta^n(\boldsymbol{X})\cos\theta^n(\boldsymbol{X}) \\ -\sin\theta^n(\boldsymbol{X})\cos\theta^n(\boldsymbol{X}) & \cos^2\theta^n(\boldsymbol{X}) \end{pmatrix} \odot \mathbf{H},
\end{aligned}
$$

where $\mathbf{H}$ is a Hessian operator and $\odot$ is a Hadamard product. The factorization of $\frac{1}{2}\sigma\sigma^T$ leads to

$$\sigma(\boldsymbol{X}) = \sqrt{2} \begin{pmatrix} -\sin\theta^n(\boldsymbol{X}) \\ \cos\theta^n(\boldsymbol{X}) \end{pmatrix} \quad \text{and by identification,} \quad \boldsymbol{b}(\boldsymbol{X}) = c(\boldsymbol{X}) \begin{pmatrix} -u_y^n(\boldsymbol{X}) \\ u_x^n(\boldsymbol{X}) \end{pmatrix}.$$

Given the functions $\boldsymbol{b}(\boldsymbol{X}(t))$, and $\sigma(\boldsymbol{X}(t))$, we come up with a pair of processes $(\boldsymbol{X}(t), B(t))$ such that the SDE in Eq. (4.15) holds. In this case, the solution $\boldsymbol{X}(t)$ is called a weak solution, as it does not specify beforehand the explicit representation of the white noise, i.e. the version $B(t)$ of the Brownian motion is not given in advance. $\qquad\square$

Both the drift and diffusion coefficient vectors of this SDE are in the tangent direction of our level curves, which helps us interpret it as a 1-dimensional Ito diffusion on the instantaneous tangent direction $\boldsymbol{T}(u_x^n(t), u_y^n(t))$. A differentiability assumption on $u(t, \boldsymbol{x})$

$$\lim_{\delta t \to 0} \frac{u(t + \delta t, \boldsymbol{x}) - u(t, \boldsymbol{x})}{\delta t} = \frac{\partial u(t, \boldsymbol{x})}{\partial t} \approx \mathcal{A}_{\text{GHElin}} u(t, \boldsymbol{x})$$

is sufficient for a short-time existence of the linearized PDE version of the nonlinear geometric heat equation.

Using Kolmogorov's theorem cited above, and assuming $u(t, \boldsymbol{x})$ and its derivatives are "sufficiently regular" (Lipschitz properties), starting at each time t, the diffusion $\boldsymbol{X}(t)$ in Eq. (4.15) is constructed for each time interval $(t - \delta t, t)$, and may be used to write a Backward Kolmogorov Equation,

$$u(t - \delta t, \boldsymbol{x}) = E\{u(t, \boldsymbol{X}(t))/\boldsymbol{X}(t - \delta t) = \boldsymbol{x}\},$$

as a mean value around each pixel dictated by the motion of the constructed diffusion $\boldsymbol{X}(t)$. This equation can also be written in forward time (since in the small time step

38

not change but only the measure on the trajectories change. This theorem involving a change of measure provides us with a means of changing the mean of the process $X(t)$ we obtained in Eq. (4.16), particularly removing the drift and obtaining the process in Eq. (4.17).

This intuitively appealing interpretation of a particle/pixel motion in the process of a diffusion is shown in the next section to be particularly useful and insightful for developing more general and feature/shape adapted flows.

4.4   A New Class of Flows

The insight gained from the tangential Brownian motion on a curve together with the normal angle $\theta(t, x)$, lead to the idea of constraining the Brownian motion at some specific orientation angles at each point x. A natural generalization of the geometric heat equation, based upon the stochastic framework presented in Section 4.3, is to construct an SDE weighted by a carefully chosen functional $h(\theta^n)$, ($h(.) \in C^\infty(\mathbb{R}^n)$) designed to capture specific features in an image, and we write locally

$$dX(t) = T^n(X(t)) \ (c^n(X(t))h(\theta^n(X(t))) \ dt + \sqrt{2} \ h(\theta^n(X(t))) \ dB(t)).$$

Here, neglecting the drift motion concentrating on pure diffusion, the Brownian motion in the tangent direction is being further constrained at some specific orientation values, i.e. at the zeros of the $h(\theta^n)$ function,

$$dX(t) \approx \sqrt{2} \ T^n(X(t)) \ h(\theta^n(X(t))) \ dB(t). \tag{4.19}$$

Constraining the diffusion of particles at points with specified orientations is aimed at extracting desired features of a contour as it is being smoothed. Such models are generated by the following class of PDE's which generalize the geometric heat flow (4.4)

$$\frac{\partial u(t, x)}{\partial t} = h^2(\theta(t, x)) \ u_{\xi\xi(t,x)}, \tag{4.20}$$

which is locally the generator of the diffusion in SDE (4.19). When applied to an image, this flow induces the following curve evolution on each iso-intensity contour $\mathcal{C}$

$$\frac{\partial \mathcal{C}}{\partial t} = h^2(\theta)\kappa N. \tag{4.21}$$

4.4.1   Well-posedness of the generalized model

Proposition 4. *The corresponding PDE's (4.20) are well-posed.*

40

Proof. The geometric heat equation which corresponds to the simplest case of this class with $h^2(\theta(t, \boldsymbol{x})) = 1$, $\forall t$, $\forall \boldsymbol{x}$, has been shown to be well-posed, and its existence and uniqueness properties may be found in [43, 47, 48].

The operator of the geometric heat equation is given by

$$L[u] = \mathcal{L}[u] - \frac{\partial u}{\partial t} = 0 \tag{4.22}$$

where

$$\mathcal{L}[u] = \sum_{i,j}^2 a_{ij} \frac{\partial^2 u}{\partial x_i \, \partial x_j} = \sin^2 \theta \; u_{xx} - 2 \sin \theta \; \cos \theta \; u_{xy} + \cos^2 \theta \; u_{yy}, \tag{4.23}$$

is the principal part of the operator L. The matrix of coefficients $[a_{ij}]$ is positive semi-definite with the eigen values 1 and 0. If we multiply this matrix by a positive function, it remains positive semi-definite. Such elliptic-parabolic operators satisfy a maximum principle (see, for example, [49]). In our case we multiply by a non-negative function $h^2(\theta)$ which can be made strictly positive by adding a very small number, $\epsilon > 0$,

$$[h^2(\theta) + \epsilon] \mathcal{L} u > 0.$$

This results in a family of nonlinear parabolic equations each of which satisfies a strong maximum principle. Our operator is obtained in the limit as $\epsilon \to 0$. $\qquad \square$

4.4.2 Polygon leading diffusions

The geometric heat equation is a rotationally invariant flow which evolves, as mentioned earlier, any shape into a circle [35]. It is the only rotationally invariant shape in Euclidean space. If we wish to capture more general shapes (triangles, squares, etc ...) it is only then natural to consider flows which are not rotationally invariant. Such a class is given by the form (4.21) when $h(\theta)$ is chosen to be other than a constant. If we are particularly interested in polygons, we may consider periodic functions (whose periodicity is dictated by the desired number of vertices) such as

$$h^2(\theta) = \begin{cases} \cos^2(n\theta) \\ \sin^2(n\theta), \end{cases} \tag{4.24}$$

leading to curve evolution equations of the form

$$\frac{\partial \mathcal{C}}{\partial t} = \cos^2(n\theta) \kappa \boldsymbol{N} \quad \text{or} \quad \frac{\partial \mathcal{C}}{\partial t} = \sin^2(n\theta) \kappa \boldsymbol{N} . \tag{4.25}$$

If we apply (4.25) to a convex shape, there will be $2n$ points on the curve which do not diffuse (corresponding to the zeros of $\cos(n\theta)$ or $\sin(n\theta)$) at equally separated rotations of the unit normal $\boldsymbol{N}$. As the unit normal moves further and further away from these angles, the diffusion increases. It hence makes sense to expect a curve to develop vertices (points of maximal curvature) at these points.

Lemma 1. *The angle of a unit normal does not change at points where the chosen function $h^2(\theta)$ vanishes. Those points, in turn, remain fixed for a short-time, and their speed remains at zero.*

Proof. Assume that a family of curves $\mathcal{C}(t, p)$, where p is any parameter along the curve, evolves according to the evolution equation

$$\frac{\partial \mathcal{C}}{\partial t} = \alpha(t, p)\boldsymbol{T} + \beta(t, p)\boldsymbol{N} \tag{4.26}$$

The evolution equation for the angle of the unit normal is given in [39] as

$$\frac{\partial \theta}{\partial t} = \frac{-1}{g}\left[\beta_p - \alpha\kappa g\right]$$

where $g = ||\mathcal{C}_p|| = \sqrt{\mathcal{X}_p^2 + \mathcal{Y}_p^2}$ is the length along the curve (metric). If we consider the case $\alpha = 0$ and $\beta = -h^2(\theta)\kappa$ (following the convention used by the authors in [39]), which corresponds to the form of the deformation we proposed, the orientation evolution is governed by

$$\begin{aligned}
\frac{\partial \theta}{\partial t} &= \frac{1}{g}\left[(h^2(\theta)\,\kappa)_p\right] \\
&= \frac{1}{g}\left\{2\,h(\theta)\,(h(\theta))_p\,\kappa + h^2(\theta)\,\kappa_p\right\} \\
&= \frac{1}{g}\,h(\theta)\left\{2\,(h(\theta))_p\,\kappa + h(\theta)\,\kappa_p\right\} \tag{4.27}
\end{aligned}$$

Notice that $\frac{\partial \theta}{\partial t} = 0$ for those points at which $h(\theta) = 0$. $\qquad\square$

We note that in [39], the orientation of a curve is defined as the angle subtended by the tangent and the x-axis, whereas we here define θ as the angle subtended by the normal and the x-axis. There is, however, complete equivalence in so far as the evolution equation of the angle θ is concerned.

In light of the above development, we can thus state that the zeros of the function $h(\theta)$ lead to fixed end points of curve segments. Fixing two end points, say a_1 and a_2, we examine the evolution of curvature, whose general form is given by (in [39])

$$\frac{\partial \kappa}{\partial t} = -\frac{\partial^2 \beta}{\partial s^2} + \alpha\frac{\partial \kappa}{\partial s} - \beta\kappa^2,$$

where s is the arc-length parameter along the curve. When substituting $\alpha = 0$ and $\beta = -h^2(\theta)\kappa$ into this equation, we have

$$\begin{aligned}
\frac{\partial \kappa}{\partial t} &= [h^2(\theta)\kappa]_{ss} + h^2(\theta)\kappa^3 \\
\frac{\partial \kappa}{\partial t} &= [(h^2(\theta))_{ss}\kappa + 2(h^2(\theta))_s\kappa_s + h^2(\theta)\kappa_{ss}] + h^2(\theta)\kappa^3 \\
\frac{\partial \kappa}{\partial t} &= \underbrace{h^2(\theta)\,\kappa_{ss}}_{\text{diffusion term}} + \underbrace{h^2(\theta)\,\kappa^3 + (h^2(\theta))_{ss}\,\kappa + 2(h^2(\theta))_s\,\kappa_s}_{\text{reaction term}}
\end{aligned}$$

42

This clearly demonstrates that a regularizing diffusion takes place, since the multiplicative factor $h^2(\theta)$ never becomes negative (which would result in an ill-posed backward diffusion). In addition, we have the reaction term which is composed of functions of κ, κ^3, and κ_s.

We have hence shown that with fixed end points, a particular curve segment subject to the new evolution equation for the curvature shown above, results in a straight line as a final solution.

Now, we can state a theorem where we put our argument of convergence to regular polygons.

Theorem 3. *A convex curve $\mathcal{C}$ subject to the evolution $\mathcal{C}_t = h^2(\theta)\kappa\boldsymbol{N}$ will converge to an M-sided, regular polygon whose M vertices will be formed at those vanishing points of the function $h^2(\theta)$.*

The proof of this theorem can be completed using the arc-length evolution equation

$$\frac{\partial L(\mathcal{C})}{\partial t} = -\int_{a_1}^{a_2} \kappa^2 h^2(\theta)ds,$$

where ds denotes the incremental arclength of $\mathcal{C}$. Since the integrand is strictly positive, we see that a curve will continue to shrink until curvature vanishes, that is the curve segment converges to a straight line between the end points a_1 and a_2. This in conjunction with the above lemma completes the proof of the theorem.

4.5   Experimental Results

4.5.1   Examples in Polygonization

To substantiate the stated theorem, we next present examples illustrated in Fig. B.5 and Fig. B.6. In our experiments with contours, we use the narrow-band implementation of the level set method developed in [50]. The time step is, $\delta t = 0.2$. Numerical implementations of the proposed flows are applied to a variety of convex shapes shown in part (a) of each figure. In Fig. B.5, the shapes in part (b) were obtained by using $h^2(\theta) = \cos^2(2\theta)$ via the following curve evolution

$$\mathcal{C}_t = \cos^2(2\theta)\kappa\boldsymbol{N}\,, \tag{4.28}$$

while the shapes in part (c) were obtained using $h^2(\theta) = \sin^2(2\theta)$. In both cases, we expect to obtain four-sided and regular polygons. The zeros of $\cos(2\theta)$ and the zeros of $\sin(2\theta)$ are however 45 degrees out of phase. As such, we see the evolved shapes in part (b) taking the form of a square, whereas the evolved shapes in part (c) take the form of a diamond, corresponding to a 45 degree rotation of the shapes in part (b). In Fig. B.6, we see the effect of using different periods. The shapes in part (b)

43

are obtained using $\mathcal{C}_t = \cos^2(3\theta)\kappa N$, while the shapes in part (c) are obtained using $\mathcal{C}_t = \sin^2(1.5(\theta - \pi/2))\kappa N$. In the first case, we expect 6 vertices, and in the second case we expect 3 vertices. Our expectations match the results shown in part (b) and (c), where we observe hexagonal and triangular shapes, respectively.

We may also apply these flows to the level sets of an image in the same manner that the geometric heat equation may be applied. This gives rise to a family of anisotropic smoothing filters which, unlike the geometric heat equation, are not rotationally invariant. This feature can be useful in smoothing noisy images where corners and edges are priorly known to have certain orientations. These diffusions are modeled by PDE's of the following form:

$$u_t = h^2(\theta)\nabla \cdot \left(\frac{\nabla u}{\|\nabla u\|}\right)\|\nabla u\|. \tag{4.29}$$

Note that the trigonometric expressions we have considered for $h^2(\theta)$ can be written in terms of the first derivatives of u, for example

$$\cos^2(2\theta) = \frac{(u_x^2 - u_y^2)^2}{(u_x^2 + u_y^2)^2} \quad \text{and} \quad \sin^2(2\theta) = \frac{(2u_x u_y)^2}{(u_x^2 + u_y^2)^2},$$

allowing one to implement the PDE without having to compute the orientation of the unit normal to each level curve. Note that these expressions involve only first order derivatives and therefore do not alter the quasi-linear structure of these second order flows.

4.5.2   Examples in Feature-Preservation

Feature-preserving properties as well as polygonal approximation properties of the proposed flows will be demonstrated in this section. We illustrate the idea of capturing different polygonal features of shapes by our proposed flows on the following examples.

The first example is a "chef" shape with both round and polygonal features as shown in Fig. B.7. Geometric heat flow $\mathcal{C}_t = \kappa N$, $(n = 0)$, evolves these features into circles as shown in the second row of Fig. B.7 for time points $t = 40, 80, 160$. Particularly, at $t = 160$, most parts of the shape turns into incomprehensible blob-like structures. In contrast to this, polygonal features of the chef like his nose, and tray, are preserved by the flow $\mathcal{C}_t = sin^2(2\theta)\kappa N$, $(n = 2)$, which favors diamond-like structures (see third row of Fig. B.7 for $t = 40, 80, 160$). Similarly, the flow $\mathcal{C}_t = cos^2(4\theta)\kappa N$, $(n = 4)$, favors octagonal features as shown on the fourth row of Fig. B.7, which is observed at chef's hat at all time points $t = 40, 80, 160$. The regularity of these flows is readily observed through the smoothness of the resulting shapes. When we view each row from left to right, we observe a progression from finer to coarser scale. The scale-spaces produced by our modified flows in the last two rows are visually more pleasing since corners are preserved, whereas in the row above we see them smoothed away by the pure geometric heat flow.

The second shape example is a fish which contains some fine detail structures as well as coarse features (Fig B.8). The second row shows the result of the geometric heat flow $\mathcal{C}_t = \kappa \mathbf{N}$, $(n = 0)$, which smoothes away not only fine features but some coarse features as well (the fins for example). The results of the flow, $\mathcal{C}_t = cos^2(2\theta)\kappa \mathbf{N}$, $(n = 2)$, are shown in the third row of Fig B.8. In this case, rectangular features are preserved for longer periods throughout the evolution. Finally, the flow, $\mathcal{C}_t = sin^2(4\theta)\kappa \mathbf{N}$, $(n = 4)$, is depicted in the last row, preserving octagonal features as shown in the nose and the dorsal things.

In the third example, we start with a noisy shape at time $t = 0$ shown in Fig. B.9. This shape is evolved with the geometric heat flow $\mathcal{C}_t = \kappa \mathbf{N}$, $(n = 0)$, the flow $\mathcal{C}_t = sin^2(1.5(\theta - \pi/2))\kappa \mathbf{N}$, $(n = 1.5)$, the flow $\mathcal{C}_t = sin^2(2\theta)\kappa \mathbf{N}$, $(n = 2)$, and the flow $\mathcal{C}_t = cos^2(2.5(\theta - \pi/2))\kappa \mathbf{N}$, $(n = 2.5)$, as shown in Fig. B.9. The geometric heat flow at the top row quickly smoothes corners of the shape out, and at coarser scales, the shape loses all of its features. The initial shape converges to a circle regardless of the global feature of the plane being a polygonal shape. This motivates the application of the geometric heat flow with a $sin^2(n\theta)$ factor, where n=1.5 whose weak limiting shape is a triangle which intuitively matches the coarser form of the given plane shape. Similarly, for $n = 2$ and $n = 2.5$, different features of the shape are kept, and persist over a much longer time period as can be observed from the column of shapes at $t = 400$. Note that the geometric heat flow result at the top quickly washes out any similarity to the actual shape, whereas the results of the other three flows keep the global shape as well as some finer details on the wings, the tail, and the head part.

4.5.3   Examples with Gray-scale Images

The proposed flows may also be applied to images in a straightforward fashion. For the case $h^2(\theta) = cos^2(2\theta)$, all level sets of the image are driven to rectangles, thereby enhancing those features in an image. Such features can be found in contemporary buildings where one example in NCSU, Centennial Campus is shown in Fig.B.10(a). The part of the building image with an additive Gaussian noise is shown in Fig.B.10(b). The 2nd row shows the results of the geometric heat flow $u_t = u_{\xi\xi}$ at $t = 10, 20, 50$. The noisy image at $t = 0$ is smoothed out very quickly at the expense of rounding off all the corners because the level sets of the image converge to circles. The 3rd row shows the $u_t = cos^2(2\theta)u_{\xi\xi}$ flow results at the same time points $t = 10, 20, 50$. Since the diffusion is constrained in order to drive image level sets to rectangles, the removal of noise is slower. However, the rectangular features still nicely appearing after noise removal (see the image on the right), makes it worthwhile to slow down the De-noising effect of the geometric heat flow according to our needs.

In Fig. B.11, an experiment involving diamond-like shapes in the image taken from a poster on a wall is shown. In the middle row, rounding effects on diamond shapes in this image performed by geometric heat flow is clearly observed during evolution. The proposed flow, shown in the bottom row, which takes the form $u_t = sin^2(2\theta)u_{\xi\xi}$ for this

particular shape is very good in carrying out a shape-adapted smoothing which takes place at the boundaries of the diamonds. The slight blurring effect on the picture at continued application however is due to the interaction between consecutive level curves.

A photograph taken by path-finder in mars, shown in Fig. B.12, is argued to be a hexagon-shaped structure on mars' surface. The particular flow adapted to this shape is given by $u_t = \cos^2(3\theta)u_{\xi\xi}$, and the resulting images at the second column of the figure demonstrate a better smoothing performance at the boundaries of the hexagon when compared to the images in the first column processed by geometric heat equation. From low scales to very large scales, the hexagon-adapted flow enhances and keeps on highlighting the related structure.

A noise contaminated Aerial image is shown in Fig. B.13(a). The geometric heat equation (see 2nd row, $t = 20, 40, 80$) sweeps away the shape information of the important details such as the city on the left bottom, the white bright rectangle on the right bottom, and the black feature at the top. The three images resulting from the $u_t = \cos^2(2\theta)\, u_{\xi\xi}$ flow, are quite sharp at the edges between both low and high contrast fields, therefore more useful in recognition of details as well as removing noise.

A last example is shown in Fig. B.14, where windows and a roof of a section of a house are seen. On the left column, are the results of the geometric heat flow $u_t = u_{\xi\xi}$ at times $t = 40, 80, 160$, and on the right are the results of $u_t = cos^2(2\theta)u_{\xi\xi}$ at same time steps. The noise is successfully removed by the geometric heat equation whose smearing effect on different regions into one another is also slow, at a cost of a problematic rounding off of corners. At time $t = 160$ for the result on the right bottom, approximately the same amount of noise as that of geometric heat equation at $t = 40$ is removed, and in addition to that the corners are still well-preserved.

4.6   Conclusions

Using insights from a stochastic formulation of the geometric heat equation, we have presented an alternative stochastic view of the nonlinear filtering with an ability to generalize and propose new flows driven by desired geometrical features, such as polygonal shapes of interest in a variety of shape recognition tasks.

Appendix A

Proof: We first proceed to re-express $\mathcal{E}(U_{n+1})$ in terms of Eq.(3.21) and $P_{n+1}(x + \delta|x) = 1 - P_{n+1}(x - \delta|x)$. By subsequently differentiating with respect to $P_n(x - \delta|x)$ and bearing in mind that a two sided-gradient is used, we have the following equation

$$
\begin{aligned}
&\partial(\mathcal{E}(U_{n+1}))/\partial P_{n+1}(x - \delta|x) \\
= \ &2[(U_n(x + \delta) - U_n(x))^2(U_{n+1}(x) - U_n(x - \delta))](\partial U_{n+1}(x))/\partial P_{n+1}(x - \delta|x) \\
+ \ &2[(U_n(x - \delta) - U_n(x))^2(U_{n+1}(x) - U_n(x + \delta))](\partial U_{n+1}(x))/\partial P_{n+1}(x - \delta|x)
\end{aligned}
$$

$$(A.1)$$

Setting $\partial(\mathcal{E}(U_{n+1}))/\partial P_{n+1}(x - \delta|x) = 0$, and assuming the non-degenerate case of $\partial U_{n+1}(x)/\partial P_{n+1}(x - \delta|x) \neq 0$, the optimal transition probability at the n-th step implies

$$
\begin{aligned}
&[(U_n(x + \delta) - U_n(x))^2 + (U_n(x - \delta) - U_n(x))^2]U_{n+1}(x) \\
= \ &(U_n(x + \delta) - U_n(x))^2 U_n(x - \delta) + (U_n(x - \delta) - U_n(x))^2 U_n(x + \delta)
\end{aligned}
\qquad (A.2)
$$

where the replacement of $U_{n+1}(x)$ with Eq.(3.21) will reduce to Eq. (3.22). Note that if $\partial U_{n+1}(x)/\partial P_{n+1}(x - \delta|x) = 0$, we can see that a left sided-gradient is equal to the right sided-gradient resulting in an optimal choice of probability $1/2$. ∎

Appendix B

Let us denote by $u^n(t, \boldsymbol{x})$ the solution to Eq (4.8):

$$\frac{\partial u^n}{\partial t} = \sin^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\ u_{xx}^n - \sin\left(2\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\ u_{xy}^n + \cos^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\ u_{yy}^n,$$

and define

$$f_1(u_x^n, u_y^n, u_{xx}^n) = \sin^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) u_{xx}^n,$$

$$f_2(u_x^n, u_y^n, u_{xy}^n) = \sin\left(2\tan^{-1}(\frac{u_y^n}{u_x^n})\right) u_{xy}^n,$$

$$f_3(u_x^n, u_y^n, u_{yy}^n) = \cos^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) u_{yy}^n.$$

If we write $u(t, \boldsymbol{x})$ as

$$u(t, \boldsymbol{x}) = u^n(t, \boldsymbol{x}) + \epsilon u(t, \boldsymbol{x})$$

then $\epsilon u(t, \boldsymbol{x})$ satisfies

$$\begin{aligned}
\frac{\partial \epsilon u}{\partial t} &= f_1(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{xx}^n + \epsilon u_{xx}) - f_1(u_x^n, u_y^n, u_{xx}^n) - \{f_2(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{xy}^n + \epsilon u_{xy}) \\
&\quad - f_2(u_x^n, u_y^n, u_{xy}^n)\} + f_3(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{yy}^n + \epsilon u_{yy}) - f_3(u_x^n, u_y^n, u_{yy}^n).
\end{aligned} \tag{B.1}$$

Assuming $f_1(\cdot, \cdot, \cdot), f_2(\cdot, \cdot, \cdot)$ and $f_3(\cdot, \cdot, \cdot)$ are differentiable in their arguments, we can expand $f_1(\cdot, \cdot, \cdot)$ in Taylor series about (u_x^n, u_y^n, u_{xx}^n), $f_2(\cdot, \cdot, \cdot)$ about (u_x^n, u_y^n, u_{xy}^n), and $f_3(\cdot, \cdot, \cdot)$ about (u_x^n, u_y^n, u_{yy}^n). For notational simplicity, let us denote by $\boldsymbol{x_1} = (u_x, u_y, u_{xx})$ and $\boldsymbol{x_1^n} = (u_x^n, u_y^n, u_{xx}^n)$, $\boldsymbol{x_2} = (u_x, u_y, u_{xy})$ and $\boldsymbol{x_2^n} = (u_x^n, u_y^n, u_{xy}^n)$, and $\boldsymbol{x_3} = (u_x, u_y, u_{yy})$ and $\boldsymbol{x_3^n} = (u_x^n, u_y^n, u_{yy}^n)$. If we assume that $\epsilon u(t, \boldsymbol{x})$ is small enough, we can neglect higher

order terms and write a linear approximation as

$$\frac{\partial \epsilon u}{\partial t} \approx \epsilon \boldsymbol{x_1} \cdot \nabla f_1(\boldsymbol{x_1^n}) - \epsilon \boldsymbol{x_2} \cdot \nabla f_2(\boldsymbol{x_2^n}) + \epsilon \boldsymbol{x_3} \cdot \nabla f_3(\boldsymbol{x_3^n}) \tag{B.2}$$

$$\frac{\partial \epsilon u}{\partial t} \approx \epsilon (u_x \ \ u_y \ \ u_{xx}) \cdot \begin{pmatrix} 2\sin\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \cos\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{-u_y^n}{(u_x^n)^2} \ u_{xx}^n \\ 2\sin\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \cos\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{1}{u_x^n} \ u_{xx}^n \\ \sin^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \end{pmatrix}$$

$$- \ \epsilon (u_x \ \ u_y \ \ u_{xy}) \cdot \begin{pmatrix} 2\cos\left(2\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{-u_y^n}{(u_x^n)^2} \ u_{xy}^n \\ 2\cos\left(2\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{1}{u_x^n} \ u_{xy}^n \\ \sin\left(2\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \end{pmatrix}$$

$$+ \ \epsilon (u_x \ \ u_y \ \ u_{yy}) \cdot \begin{pmatrix} 2\cos\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\left(-\sin\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\right)\frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{-u_y^n}{(u_x^n)^2} \ u_{yy}^n \\ 2\cos\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\left(-\sin\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right)\right)\frac{1}{1+(\frac{u_y^n}{u_x^n})^2} \frac{1}{u_x^n} \ u_{yy}^n \\ \cos^2\left(\tan^{-1}(\frac{u_y^n}{u_x^n})\right) \end{pmatrix} \tag{B.3}$$

Defining the corresponding nominal angle $\theta^n(t, \boldsymbol{x}) = \tan^{-1}(\frac{u_y^n(t,\boldsymbol{x})}{u_x^n(t,\boldsymbol{x})})$, and re-arranging the terms of Eq. (B.3), we get the linearized version of the geometric heat equation around a nominal value:

$$\frac{\partial u(t, \boldsymbol{x})}{\partial t} \approx \mathcal{A}_{\text{GHElin}} u(t, \boldsymbol{x}) = \sin^2(\theta^n(t, \boldsymbol{x})) \ u_{xx}(t, \boldsymbol{x}) - \sin(2\theta^n(t, \boldsymbol{x})) \ u_{xy}(t, \boldsymbol{x}) +$$
$$\cos^2(\theta^n(t, \boldsymbol{x})) \ u_{yy}(t, \boldsymbol{x}) + c(-u_y^n(t, \boldsymbol{x}) \ u_x(t, \boldsymbol{x}) + u_x^n(t, \boldsymbol{x}) \ u_y(t, \boldsymbol{x})) \tag{B.4}$$

where $c = \frac{1}{(u_x^n)^2+(u_y^n)^2} \left[\sin(2\theta^n)(u_{xx}^n - u_{yy}^n) - \cos(2\theta^n)2u_{xy}^n\right]$.
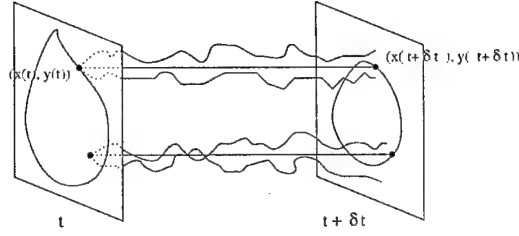
Figure B.1: Points of the zero-level set, i.e. initial contour $(\mathcal{X}(t), \mathcal{Y}(t))$, at time t, is shown on the left. Those points whose sample realizations result in an average value of zero at time $t + \delta t$ $(u(t + \delta t, \boldsymbol{x}) = E^{\boldsymbol{x}}[u(\boldsymbol{X}(t))] = 0)$ form the new contour$(\mathcal{X}(t + \delta t), \mathcal{Y}(t + \delta t))$ (on the right).
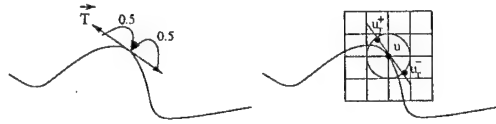
Figure B.2: Brownian Motion on the tangent direction, and corresponding interpolation on square grid.
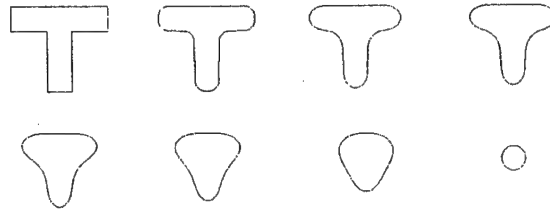
Figure B.3: Equivalent random walk on the tangent direction implemented on the level set function $u(x, y)$. The tangent direction is estimated directly from the level set set function : $\theta_T = \tan^{-1}\left(\frac{-u_x}{u_y}\right)$ The level set function is on a 250×250 grid, $\delta t = 0.25$.
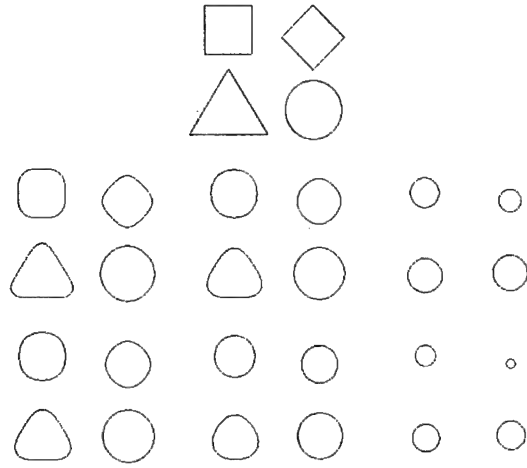
Figure B.4: Middle row: Brownian Motion on $\vec{T}$ is shown to produce similar results with those in Bottom Row: Geometric Heat flow. The speeds of the two algorithms are different. The level set function is on a 191×221 grid, $\delta t = 0.25$.
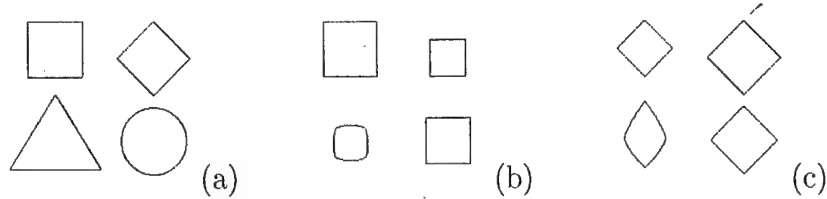
Figure B.5: (a) Initial set of shapes (b) Flow $C_t = \cos^2(2\theta)\kappa N$ (c) Flow $C_t = \sin^2(2\theta)\kappa N$.
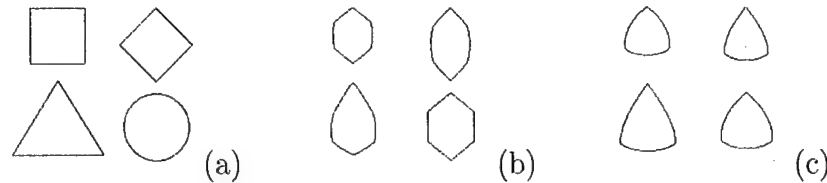
Figure B.6: (a) Initial set of shapes (b)Flow $C_t = \cos^2(3\theta)\kappa N$, which tends to produce hexagons, (c)Flow $C_t = \sin^2(1.5(\theta - \pi/2))\kappa N$, which tends to produce triangle-like shapes.
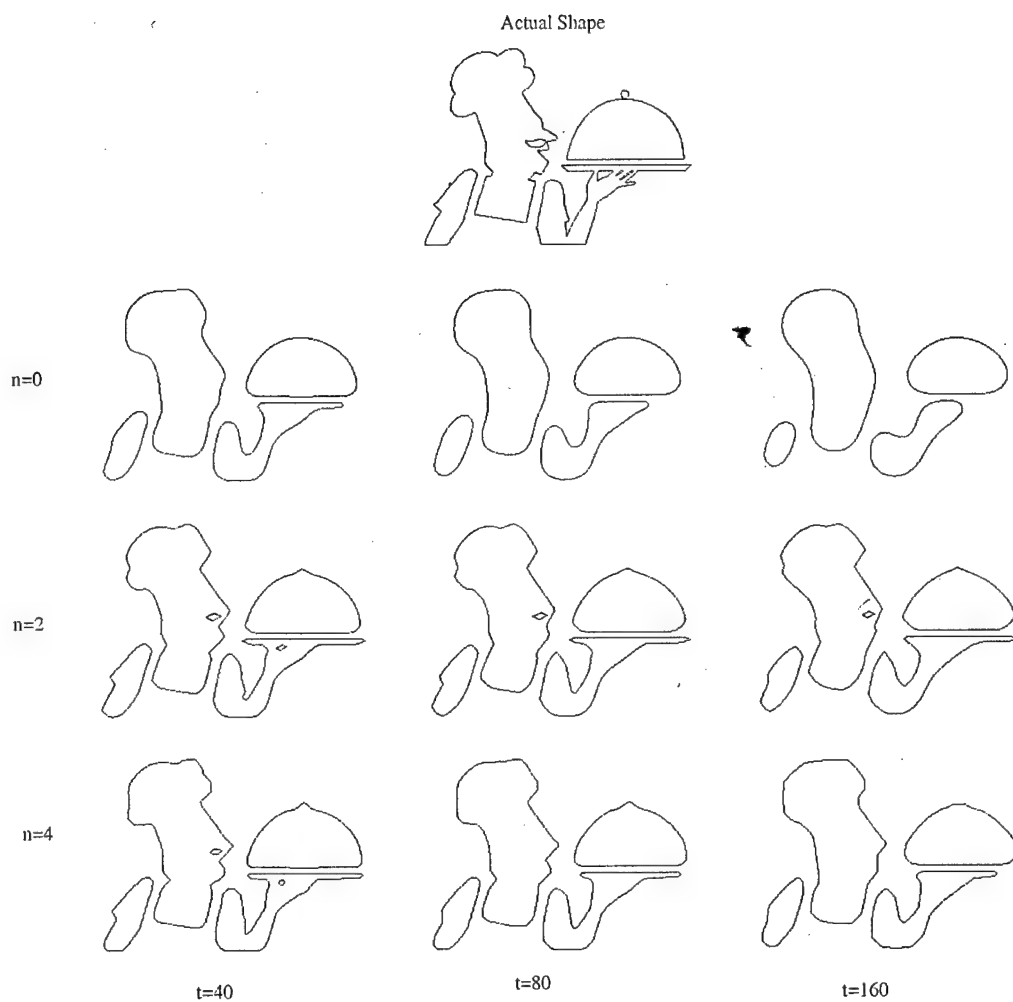
Figure B.7: Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \boldsymbol{N}$, 2nd row: $\mathcal{C}_t = sin^2(2\theta)\kappa \boldsymbol{N}$, 3rd row: $\mathcal{C}_t = cos^2(4\theta)\kappa \boldsymbol{N}$.
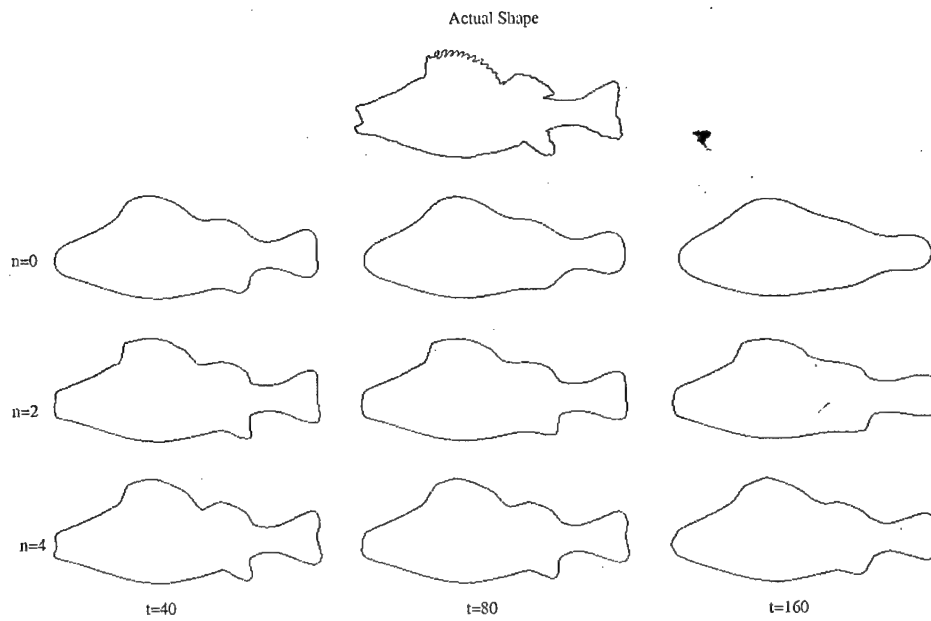
Figure B.8: Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \mathbf{N}$, 2nd row: $\mathcal{C}_t = cos^2(2\theta)\kappa \mathbf{N}$, 3rd row: $\mathcal{C}_t = sin^2(4\theta)\kappa \mathbf{N}$.

t=50   t=100   t=200   t=400

n=0

Actual Shape

n=1.5

Noisy Shape

n=2

n=2.5

54
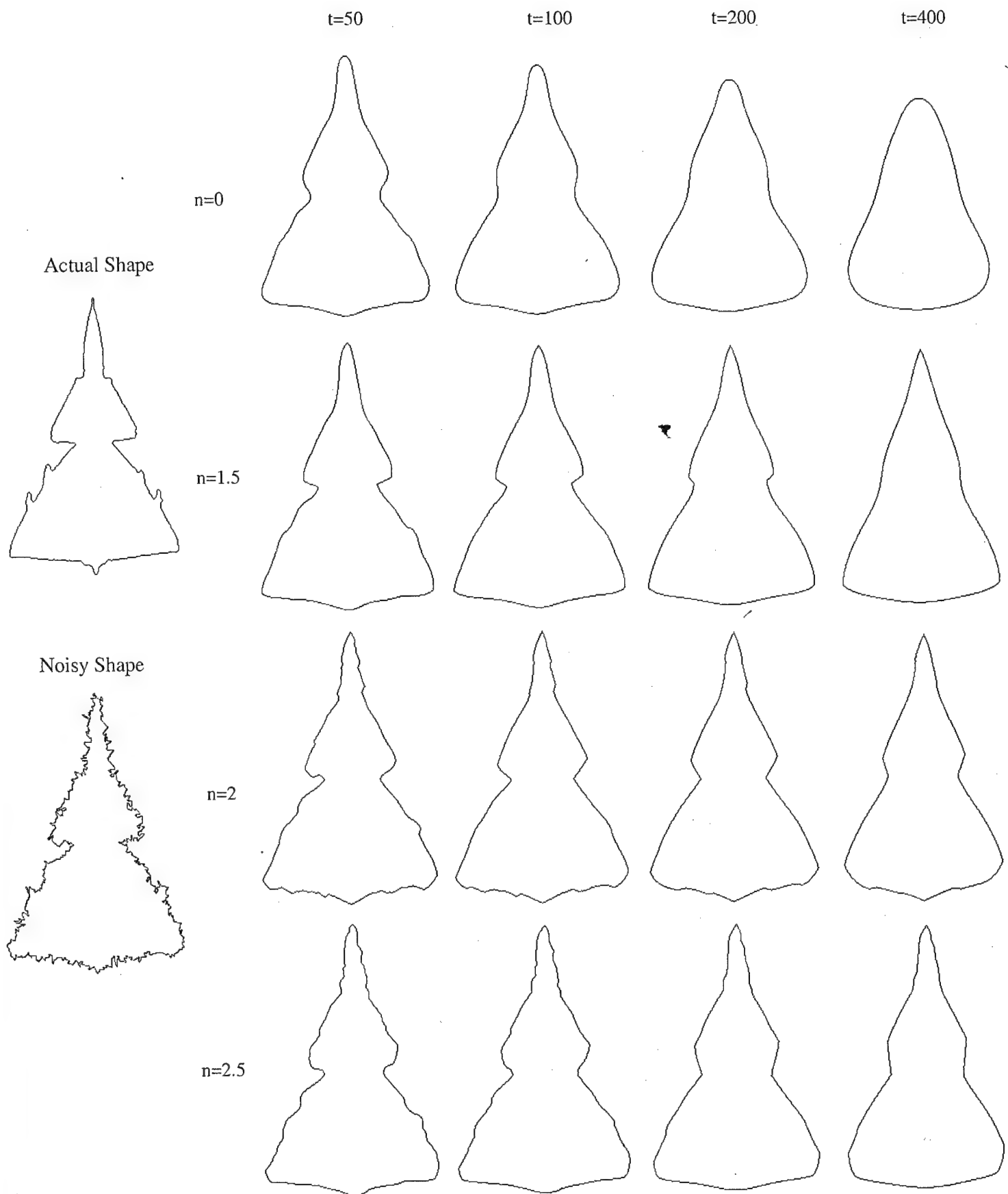
Figure B.9: Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \mathbf{N}$, 2nd row: $\mathcal{C}_t = sin^2(1.5(\theta - \pi/2))\kappa \mathbf{N}$, 3rd row: $\mathcal{C}_t = sin^2(2\theta)\kappa \mathbf{N}$, 4th row: $\mathcal{C}_t = cos^2(2.5(\theta - \pi/2))\kappa \mathbf{N}$
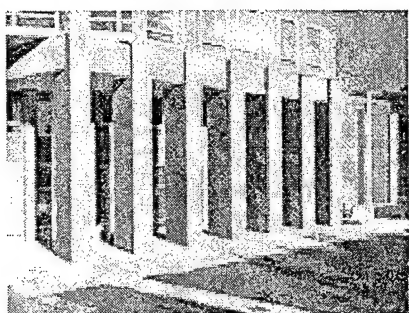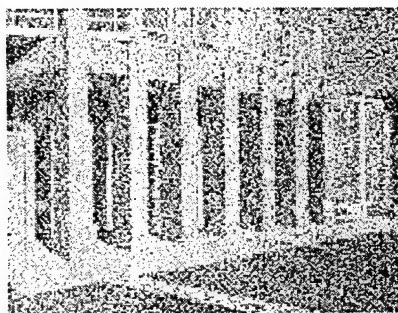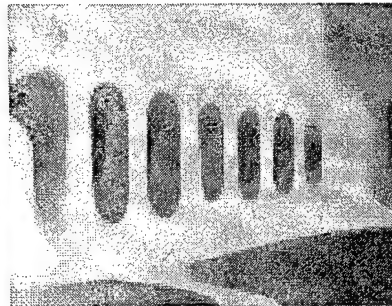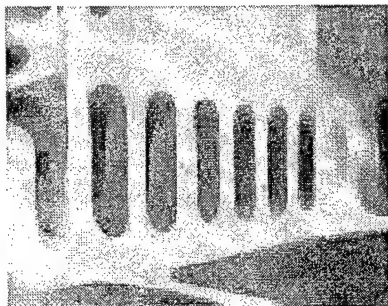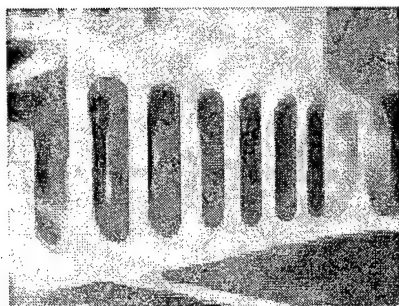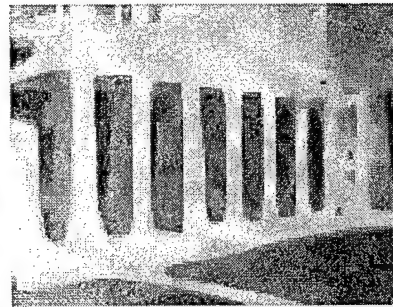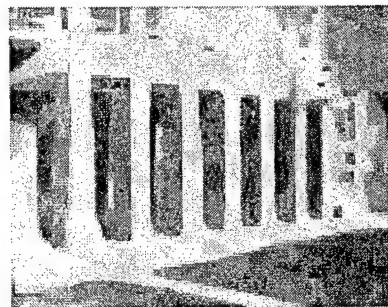
Figure B.10: (a) Clean building image (b) noisy building image (c) Geometric heat flow $u_t = u_{\xi\xi}$ (left-right) $t = 10, 20, 40$ (d) Flow $u_t = \cos^2(2\theta)\, u_{\xi\xi}$ (left-right) $t = 10, 20, 40$.
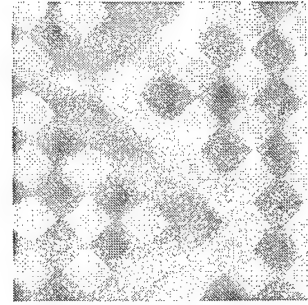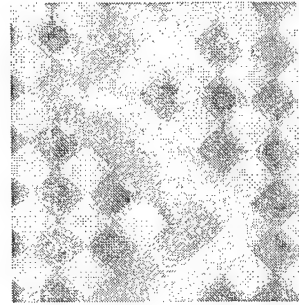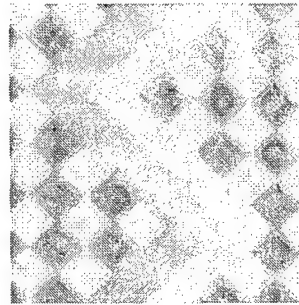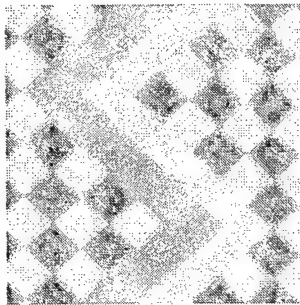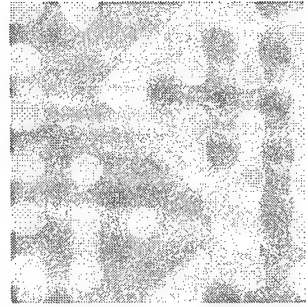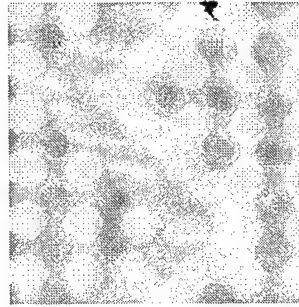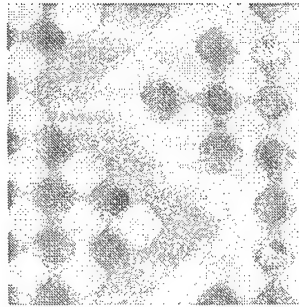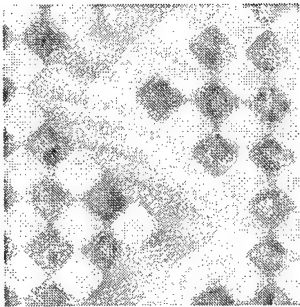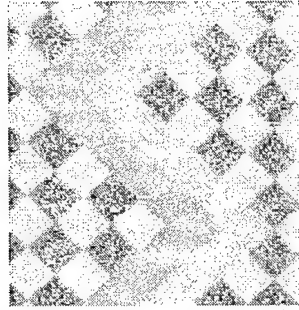
t=2   t=5   t=10   t=20

Figure B.11: (Top) Diamonds image (Middle row) Geometric heat flow $u_t = u_{\xi\xi}$ (Bottom row) Flow $u_t = \sin^2(2\theta)\, u_{\xi\xi}$.
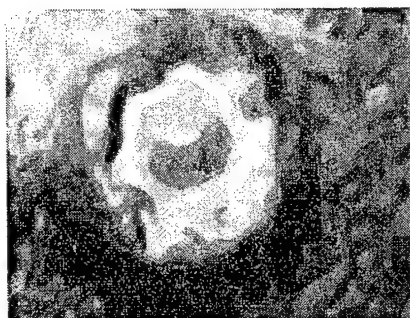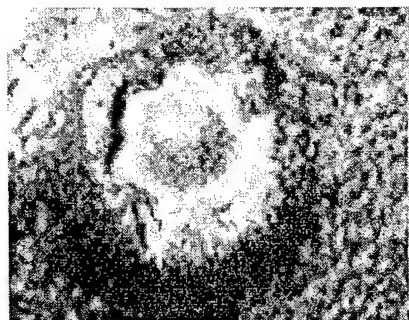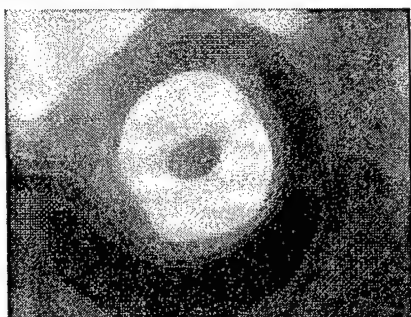
t=12.5

t=25
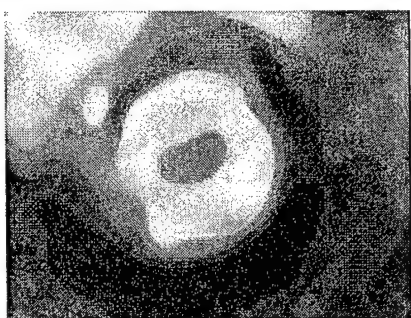
t=50

t=90

Figure B.12: (Top) An image from mars pathfinder, (First column) Geometric heat flow $u_t = u_{\xi\xi}$, (Second column) Flow $u_t = \cos^2(3\theta)\, u_{\xi\xi}$.
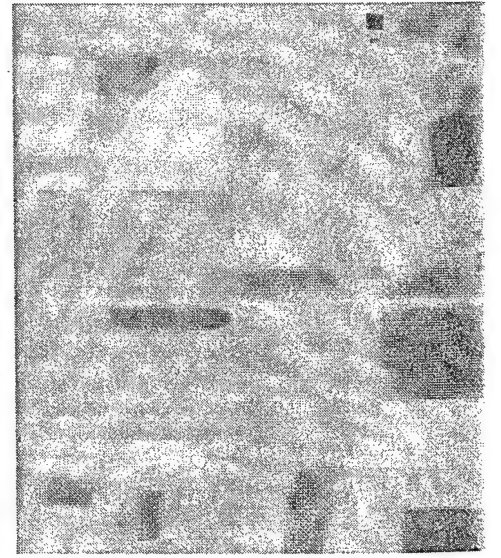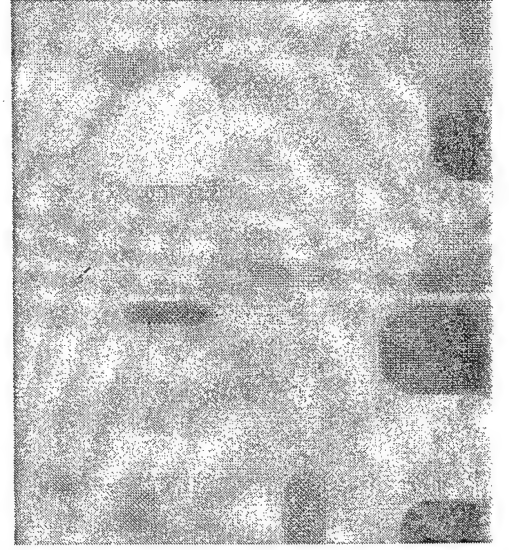
Figure B.13: Top: Aerial image; 2nd row: geometric heat flow $u_t = u_{\xi\xi}$, (left to right)$t = 20, 40, 80$; 3rd row: flow $u_t = cos^2(2\theta)u_{\xi\xi}$, (left to right)$t = 20, 40, 80$.
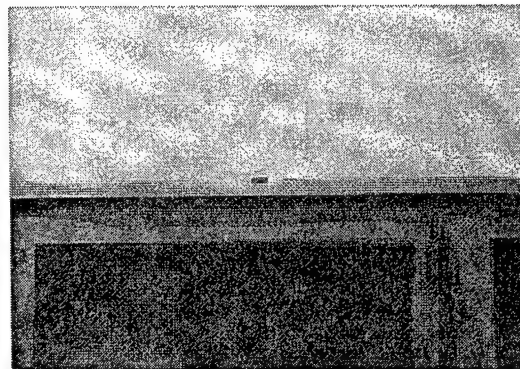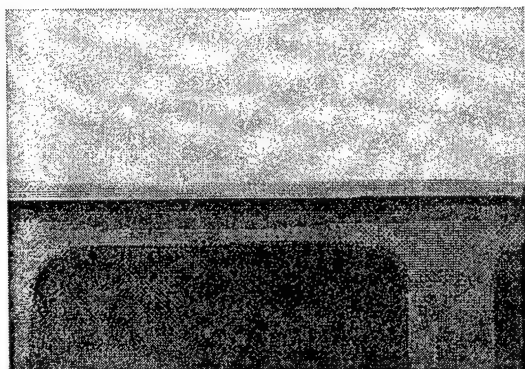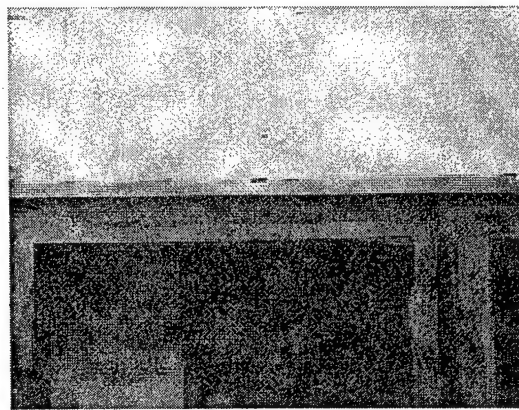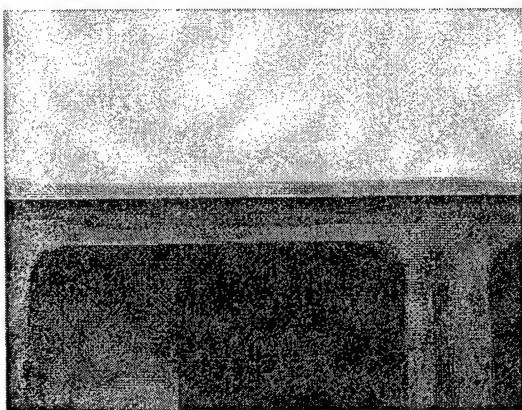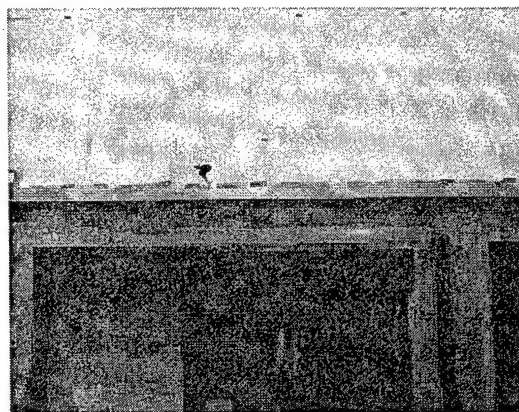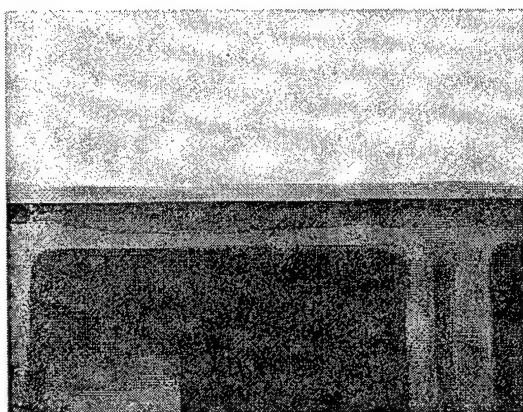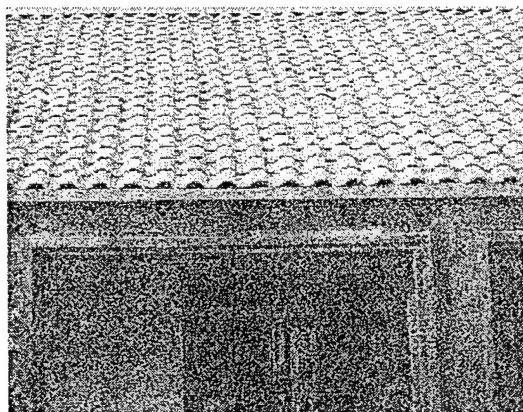
Figure B.14: Top: House; Left: Geometric heat flow $u_t = u_{\xi\xi}$, (top to bottom) $t = 40, 80, 160$; Right: Flow $u_t = \cos^2(2\theta)\, u_{\xi\xi}$, (top to bottom) $t = 40, 80, 160$.

Bibliography

[1] B. Karacali, "Signal reconstruction using the cross scale correlation properties of the wavelet decomposition," Master's thesis, North Carolina State University, July, 1999.

[2] H. Krim and I. Schick, "Minimax description length for signal denoising and optimized representation," *IEEE Trans. on Information Theory*, April 1999. IEEE Trans. on IT Special Issue, Eds. H. Krim, W. Willinger, A. Iouditski and D. Tse.

[3] H. Krim, D. Tucker, S. Mallat, and D. Donoho, "Near-optimal risk for best basis search," *IEEE Trans. on Information Theory*, Nov., 1999.

[4] A. Kim and H. Krim, "Hierarchical stochastic modeling of sar imagery for segmentation/compression," *IEEE Trans. on SP*, Feb. 1999.

[5] I. Pollak, A. Willsky, and H. Krim, "Image segmentation and edge enhancement with stabilized inverse diffusion equations," *IEEE Trans. on Image Processing*, Feb, 2000.

[6] H. Krim and Y. Bao, "Smart nonlinear diffusion:a probabilistic approach," *Submitted to IEEE Trans. on PAMI*.

[7] G. Bozkurt, H. Krim, and A. Yezzi, "Stochastic differential equations and geometric flows," *submitted to International Journal on Computer Vision*, 2000.

[8] A. B. Hamza and H. Krim, "Relaxed minimax approach to image denoising," *submitted to IEEE Trans. on SP*, 00.

[9] H. Krim, *Wavelets in Statistical Signal Processing*. Encyclopedia in Telecommunications, Wiley, 2000.

[10] H. Krim, *Stochastic Nonlinear Diffusion and Multiscale Analysis*. SIAM, 2000.

[11] H. Krim and Y. Bao, "A controlled diffusion approach to signal denoising," in *ICASSP'99, Phoenix,AZ, 1999.*, 1999.

[12] H. Krim and Y. Bao, "Stochastic diffusion in signal analysis," in *ICIP, Kobe, Japan, 1999.*, 1999.

[13] G. Unal, H. Krim, and A. Yezzi, "A stochastic flow for feature extraction," in *ICASSP*, (Istanbul, Turkey), June 2000.

[14] G. Unal, H. Krim, and A. Yezzi, "Feature driven stochastic flow," in *Proc. IEEE Int. Conf. on Image Processing*, (Vancouver, CA), September 2000.

[15] E. D. Kaplan, ed., *Understanding GPS, Principles and Applications*. Boston: Artech House, 1996.

[16] B. Karacali, "Signal reconstruction using the cross scale correlation properties of the wavelet decomposition," Master's thesis, North Carolina State University, 1999.

[17] A. P. Witkin, "Scale space filtering," in *Proc. Int. Joint Conf. on Artificial Intellingence*, (Karlsruhe, Germany), pp. 1019–1023, 1983.

[18] J. J. Koendrink, "The structure of images," *Biol. Cybern.*, vol. 50, pp. 363–370, 1987.

[19] A. L. Yuille and T. Poggio, "Scaling theorems for zero-crossings," *JOSA*, vol. A, May 1985.

[20] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. PAMI-11, pp. 674–693, Jul. 1989.

[21] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage." preprint Dept. Stat., Stanford Univ., Jun. 1992.

[22] H. Krim and J.-C. Pesquet, *On the Statistics of Best Bases Criteria*, vol. Wavelets in Statistics of *Lecture Notes in Statistics*. Springer-Verlag, July 1995.

[23] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Signal estimation using wavelet-markov models," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing — ICASSP '97*, (Munich, Germany), 1997.

[24] P. Perona and J. Malik, "A network for multiscale image segmentation," in *IEEE Int. Symp. on Circuits and Systems*, (Helsinki), pp. 2565–2568, June 1988.

[25] B. ter Haar Romeney, *Geometry Driven Diffusion*. Dordecht: Kluwer Academic Publishers, 1994. Edited Book.

[26] G. Sapiro and A. Tannenbaum, "Affine invariant scale-space," *International Journal of Computer Vision*, vol. 11, pp. 25–44, 1993.

[27] L. Alvarez, F. Guichard, P. Lions, and J. Morel, "Axioms and fundamental equations in image processing," *Arch. for Rational Mechanics*, vol. 123, pp. 199–257, Sept. 1993.

[28] H. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*. Applications of Mathematics, Springer-Verlag, first ed., 1992.

[29] B. Oksendal, *Stochastic Differential Equations: An Introduction with Applications.* Berlin: Springer-Verlag, 1992.

[30] R. Durrett, *Stochastic Calculus: A Practical Introduction.* Florida: CRC Press, first ed., 1997.

[31] *Controlled Markov Processes.* Series of Comprehensive Studies in Mathematics, Springer-Verlag, 1975.

[32] A. Papoulis, *Probability, Random Variables and Stochastic Processes.* Mc Graw Hill, 1984.

[33] S. Osher and J. Sethian, "Fronts propagating with curvature dependent speed: algorithms based on the hamilton-jacobi formulation," *J. Computational Physics*, vol. 49, pp. 12–49, 1988.

[34] J. Sethian, *Level Set Methods and fast marching methods.* Cambridge University Press, 1999.

[35] M. Grayson, "The heat equation shrinks embedded plane curves to round points," *J. Diff. Geom.*, vol. 26, pp. 285–314, 1987.

[36] H. Krim and Y. Bao, "Smart nonlinear diffusion: A probabilistic approach," Submitted to PAMI, 2000.

[37] A. Bruckstein, G. Sapiro, and D. Shaked, "Evolutions of planar polygons," *Int. J. Pattern Recognition and Artificial Intell.*, vol. 9, no. 6, pp. 991–1014, 1995.

[38] A. Witkin, "Scale space filtering," in *Proc. Int. Conf. on AI*, (Karlsruhe, Germany), pp. 1019–1023, 1983.

[39] B. Kimia, A. Tannenbaum, and S. Zucker, "On the evolution of curves via a function of curvature, I the classical case," *J.Math. Analysis and Applications*, vol. 163, pp. 438–458, 1992.

[40] B. Kimia and K.Siddiqi, "Geometric heat equation and nonlinear diffusion of shapes and images," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1994.

[41] B. Kimia, A. Tannenbaum, and S. Zucker, "Shapes, shocks, and deformations I," *Int. Journal of Computer Vision*, no. 15, pp. 189–224, 1995.

[42] L. Alvarez, F. Guichard, P. Lions, and J. Morel, "Axioms and fundamental equations of image processing," *Arch. Rational Mech. Anal.*, vol. 123, pp. 199–257, 1993.

[43] L. Alvarez, P. Lions, and J. Morel, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, vol. 29, no. 3, pp. 845–866, 1992.

[44] G. Sapiro and A. Tannenbaum, "Affine invariant scale space," *Int. Journal of Computer Vision*, vol. 11, pp. 25–44, 1993.

[45] G. Sapiro and A. Tannenbaum, "On invariant curve evolution and image analysis," *Indiana Univ. Journal of Math.*, vol. 93, 1993.

[46] B. Oksendal, *Stochastic Differential Equations, An introduction with applications.* Berlin: Springer, 1998.

[47] M. Crandall, H. Ishii, and P. Lions, "User's guide to viscosity solutions of second order partial differential equations," *Bull. Amer. Math. Soc.*, vol. 27, no. 1, pp. 1–67, 1992.

[48] Y.-G. Chen, Y. Giga, and S. Goto, "Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations," *J. Diff. Geom.*, vol. 33, pp. 749–786, 1991.

[49] J. Rauch, *Partial Differential Equations.* New York: Springer-Verlag, 1991.

[50] D. Adalsteinsson and J. Sethian, "A fast level set method for propagating interfaces," *J. Computational Physics*, vol. 118, pp. 269–277, 1995.